Multi-Paradigm Modelling (MPM)

a “Most Appropriate” Talk

Hans Vangheluwe

The fourth Model Based Space Systems and Software Engineering workshop (MBSE 2023)
on reducing the gap between model-based systems engineering
and domain-specific model-based approaches.

ESTEC, Noordwijk, The Netherlands.
16 November 2023.
Satellite of ADCSS2023.

Universiteit Antwerpen
Ansymo
Antwerp Systems & Software Modelling
University of Antwerp
Nexor
Cyber-Physical Systems
University of Antwerp
FLANDERS MAKE
MANUFACTURING INNOVATION NETWORK
McGill
MPM4CPS
cost

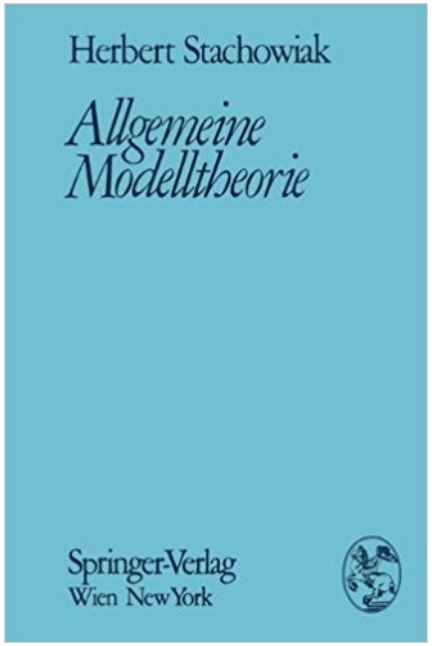

MODEL EVERYTHING!
04/04 Bellairs, Barbados
04/05 Bellairs, Barbados
04/06 Bellairs, Barbados
10/06 Genoa, Italy
04/07 Bellairs, Barbados
10/07 Nashville, TN, USA
04/08 Bellairs, Barbados
04/09 Bellairs, Barbados
10/09 Denver, CO, USA
04/10 Bellairs, Barbados
10/10 Oslo, Norway
04/11 Bellairs, Barbados
10/11 Wellington, NZ
04/12 Bellairs, Barbados
10/12 Innsbruck, Austria
05/13 Bellairs, Barbados
09/13 Miami, FL, USA
02/14 Bellairs, Barbados
09/14 Valencia, Spain
01/15 Bellairs, Barbados
09/15 Ottawa, Canada
04/16 Bellairs, Barbados
03/17 Bellairs, Barbados
05/18 Bellairs, Barbados
04/19 Bellairs, Barbados
09/19 Munich, Germany
10/20 Montreal, Canada*
10/21 Fukuoka, Japan*
04/22 Bellairs, Barbados
10/22 Montreal, Canada
03/23 Carghjese, Corsica
05/23 Bellairs, Barbados
10/23 Västerås, Sweden
*virtual event
http://CAMPaM.MPM4CPS.eu

Context: Engineering of CPS

Truly complex, engineered systems, known as **Cyber Physical Systems (CPS)**, are becoming increasingly common. CPS emerge from the **networking** of multi-**physical** (mechanical, electrical, hydraulic, biochemical, ⟩.) and **computational** (control, signal processing, logical inference, planning, ...) processes, often interacting with a highly uncertain **environment**, including **human** actors, in a **socio-economic context**.

MPM4CPS

cost
EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

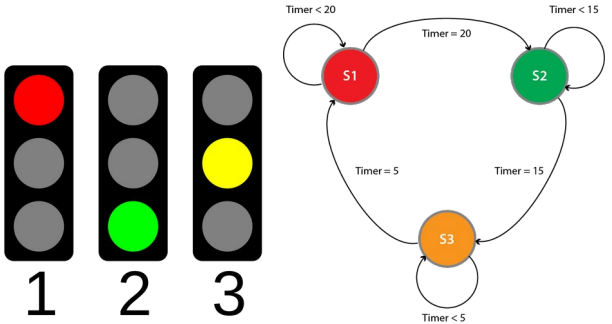
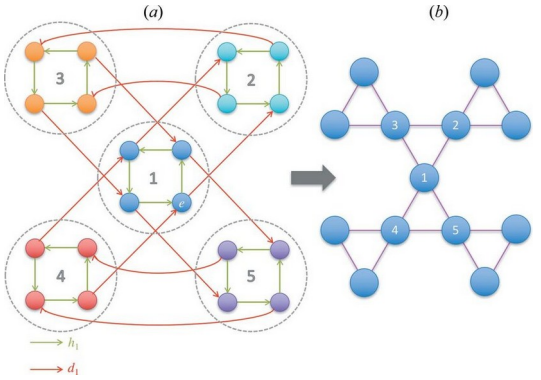


1973

“Model” Features

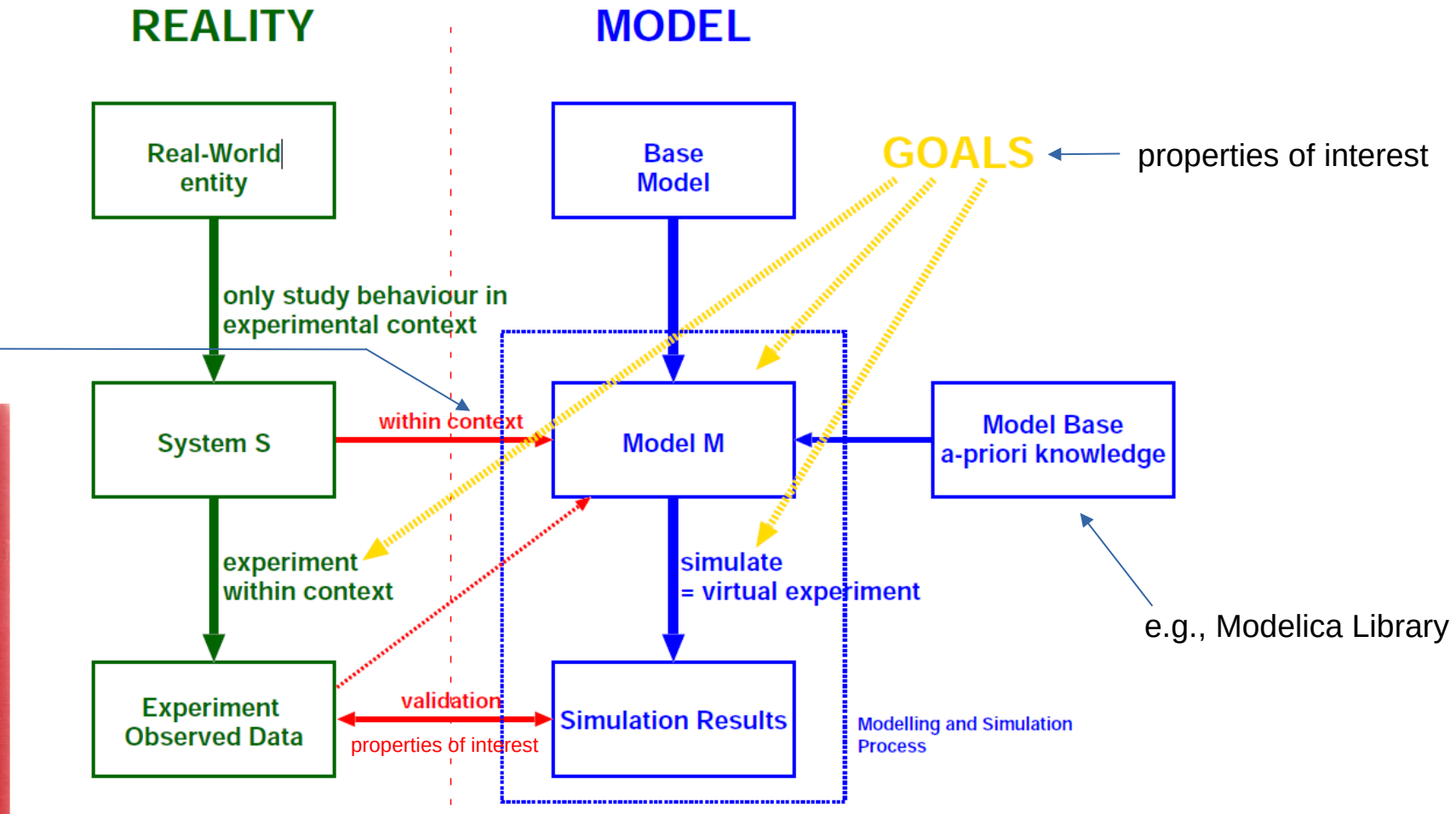
mapping feature	A model is based on an original.[4]
reduction feature	A model only reflects a (relevant) selection of an original’s properties.
pragmatic feature	A model needs to be usable in place of an original with respect to some purpose.

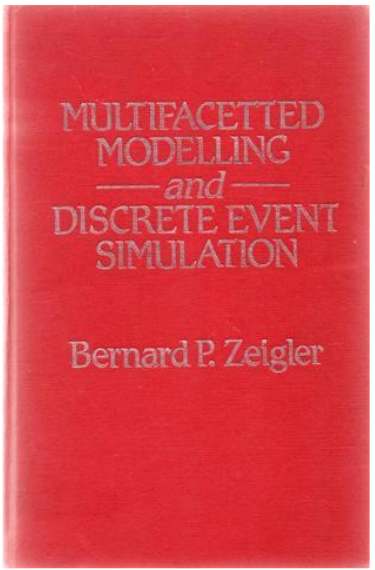
System under Study (SuS) vs. **Appropriate Model**

	Real-World Model	Virtual Model
Real-World SuS		
Virtual SuS	 	

A **Valid** Model is an **Appropriate** Model

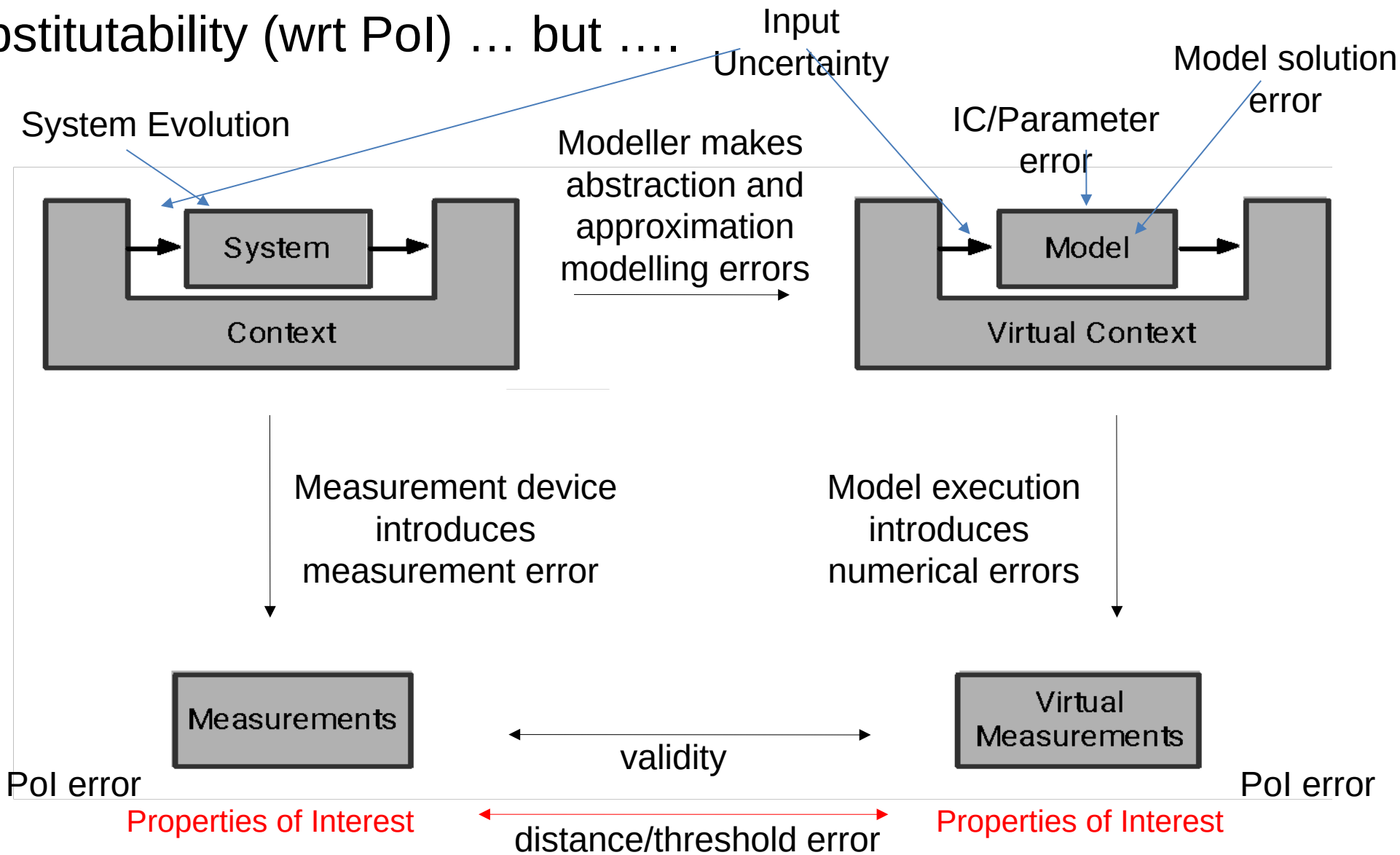
purpose: substitutability (engineering), explainability (science)



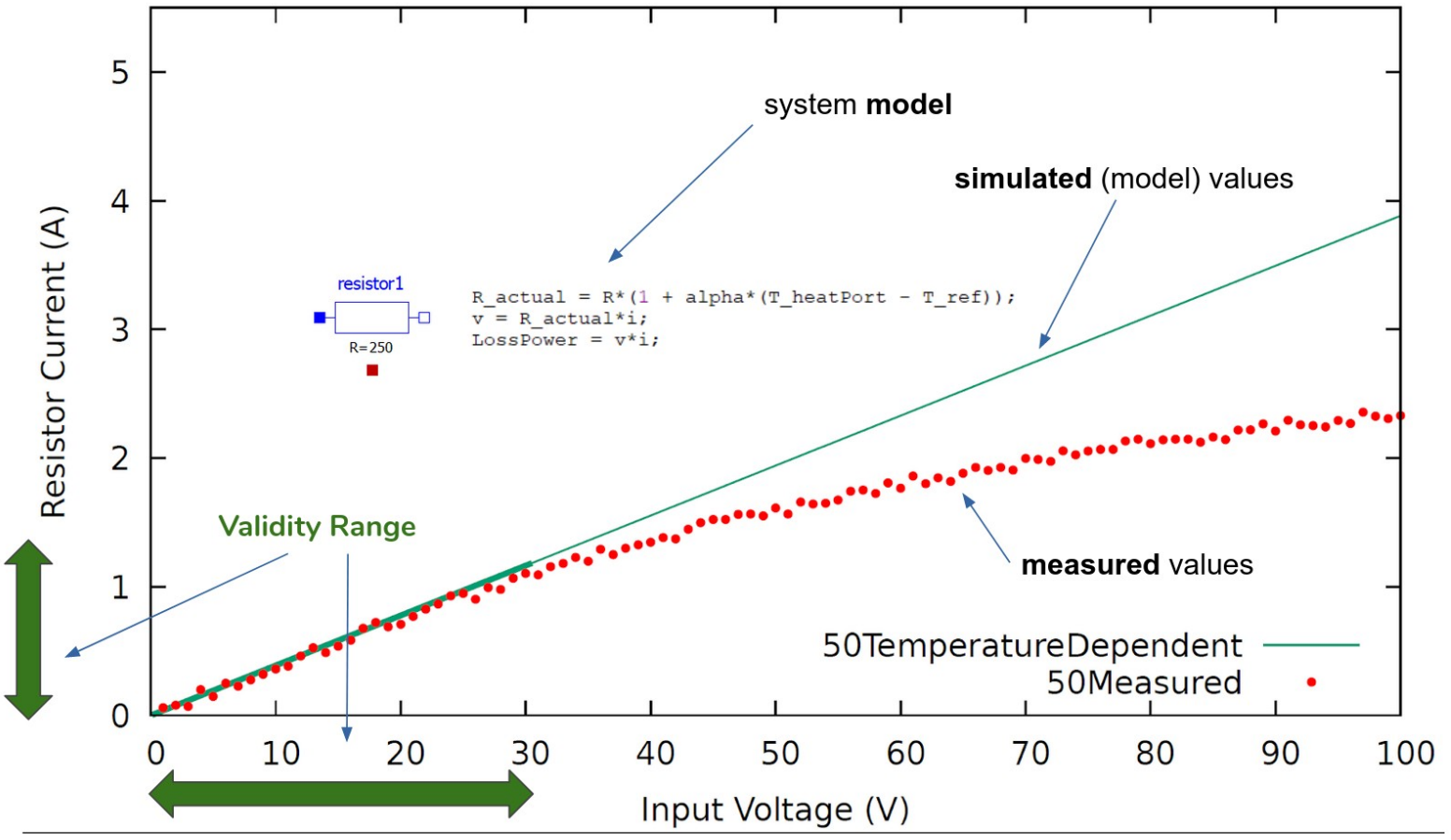


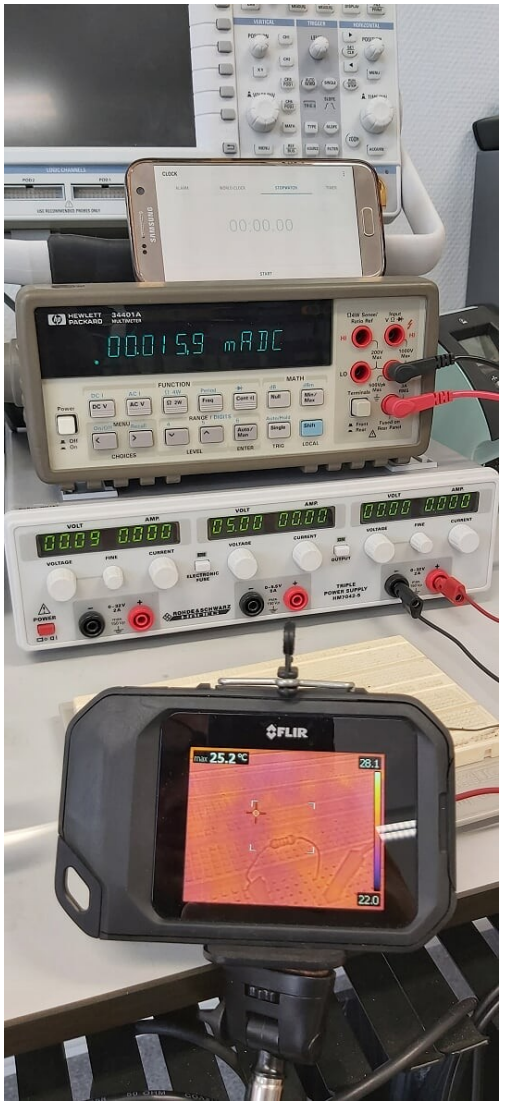
Bernard P. Zeigler. *Multi-faceted Modelling and Discrete-Event Simulation.* Academic Press, 1984.

Substitutability (wrt PoI) … but ….

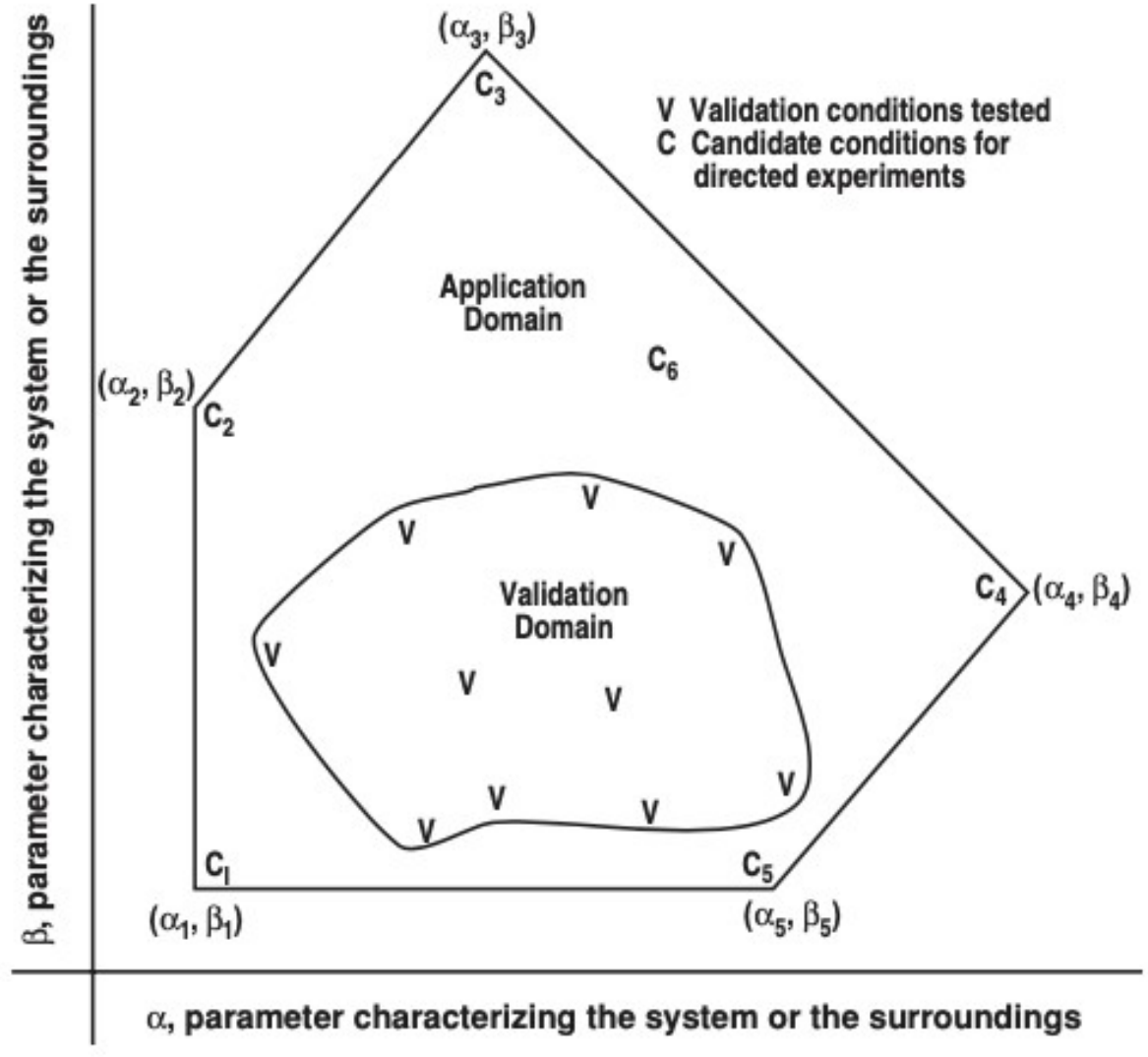


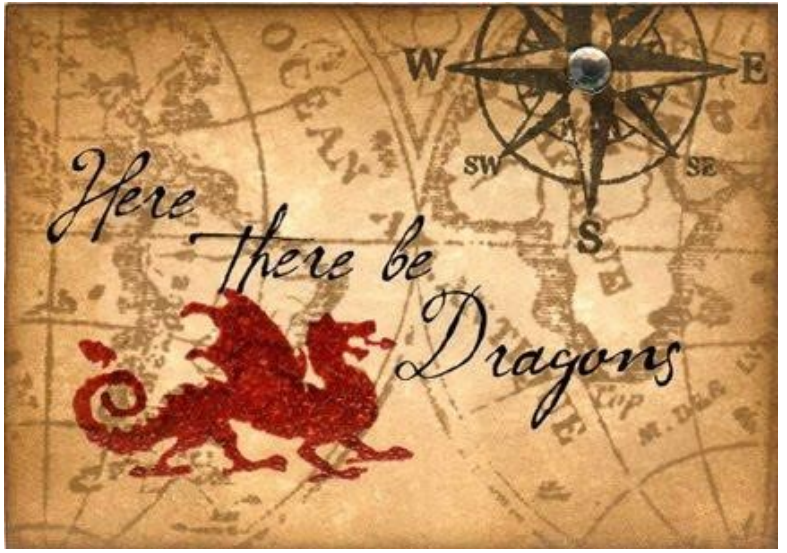
A Resistor Model's Validity Range





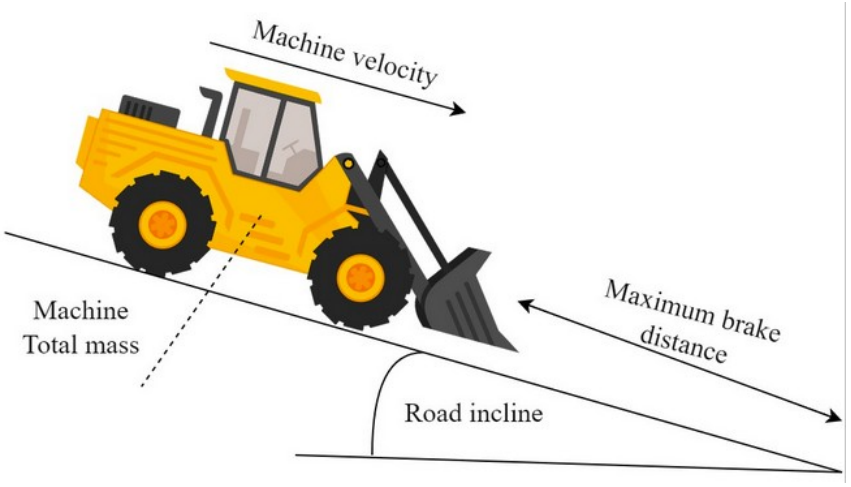
https://msdl.uantwerpen.be/cloud/public/fcfc42

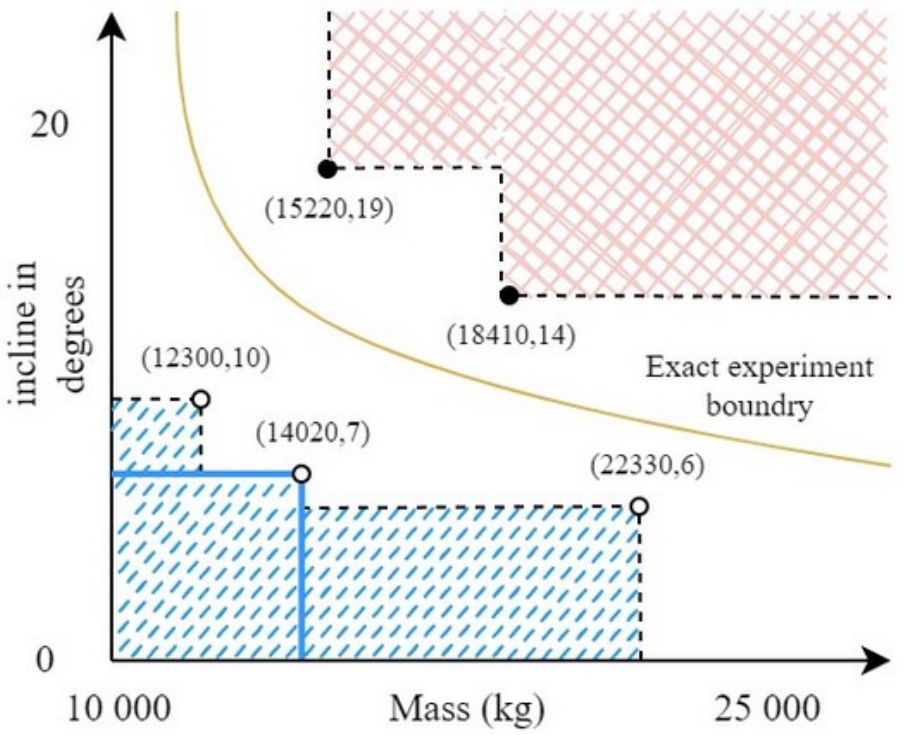




W. Oberkampf, C. Roy. Verification and Validation in Scientific Computing. Cambridge University Press, 2010.

Inferred Concrete (In)Validity Frame





Johan Cederbladh, Loek Cleophas, Eduard Kamburjan, Lucas Lima and Hans Vangheluwe. Symbolic Reasoning for Early Decision-Making in Model-Based Systems Engineering . In the 1st Workshop on Model-based Systems Engineering, MoDELS 2023.

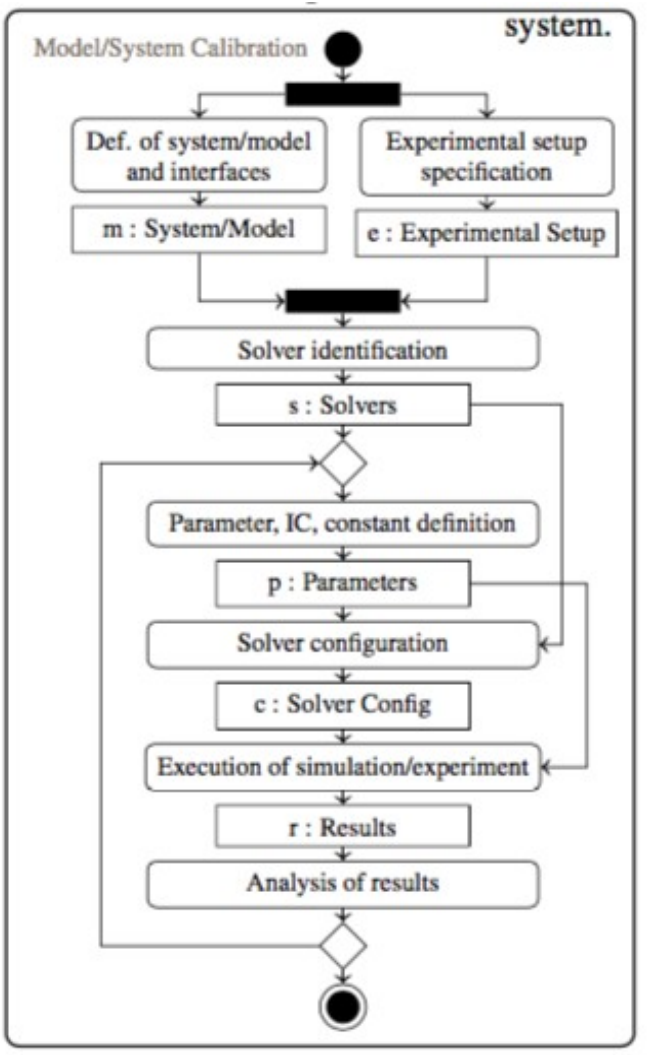
Concrete Validity Frame

must be modelled, managed, extended, evolved, re-used, …

Experiments (architecture and **worklow**):

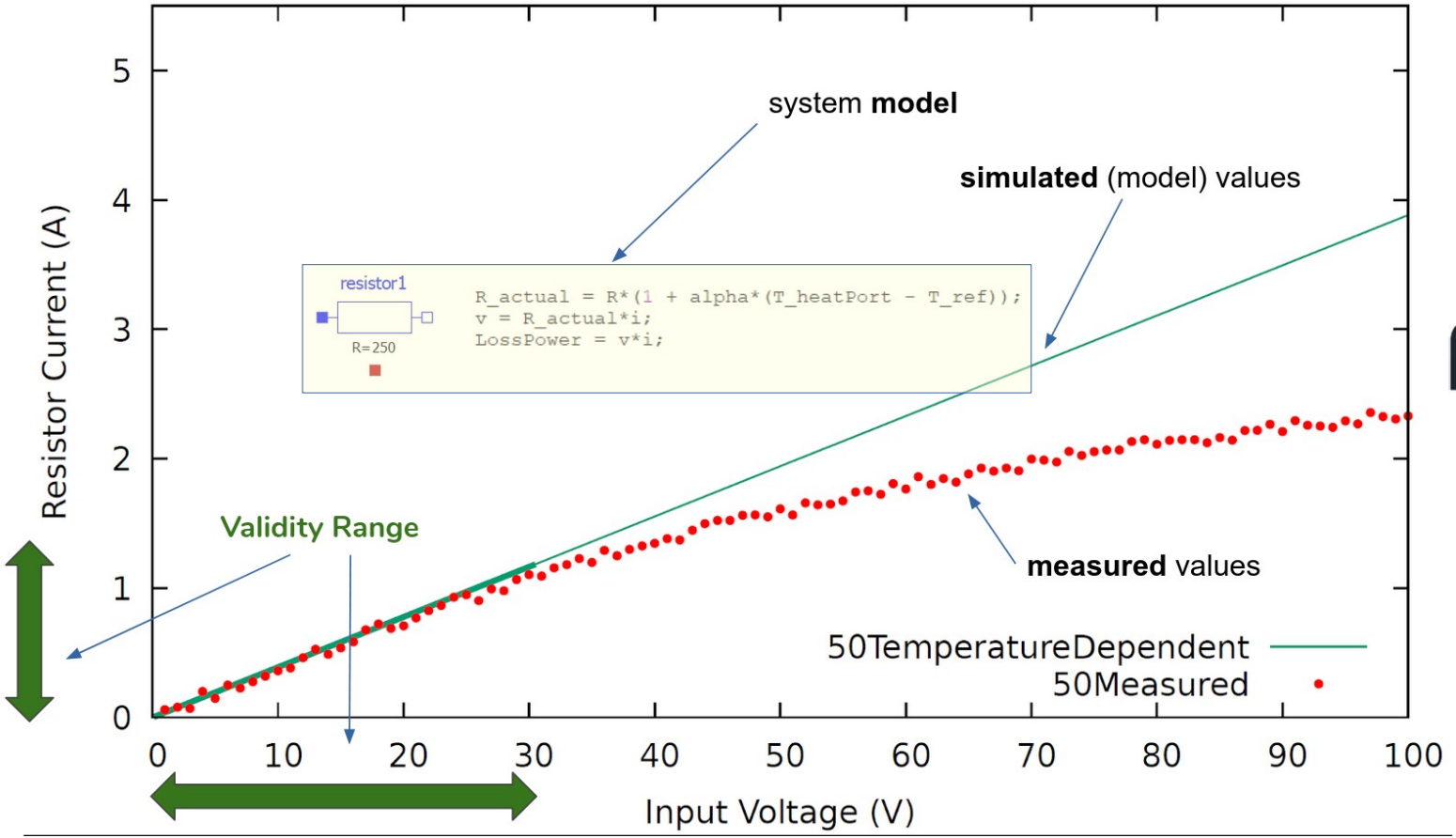
Repeatable
Replicable
Reproducible

Validity vs. Accuracy vs. Fidelity …



A Resistor Model's Validity Range

appropriate language?

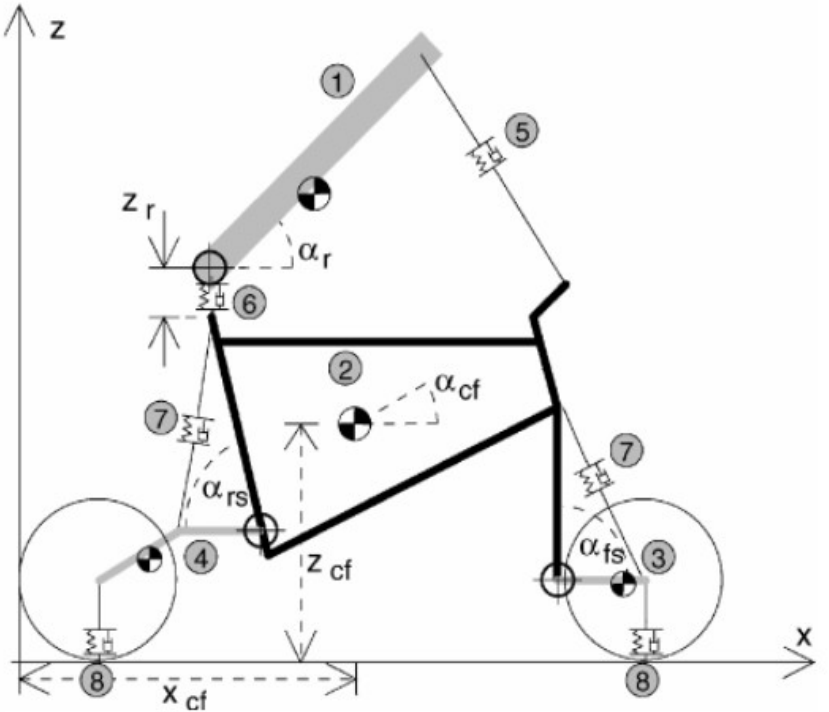


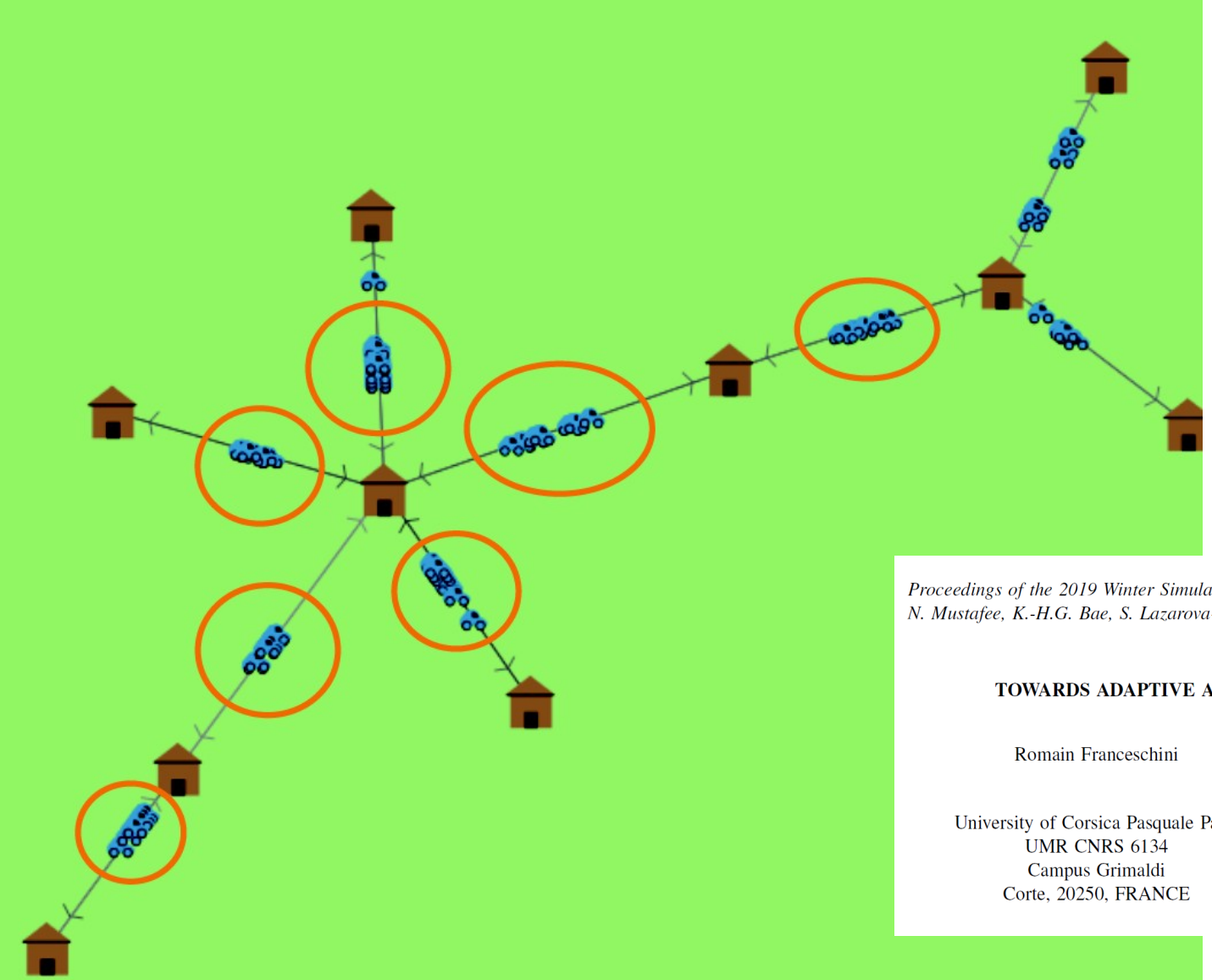
Modelica

Most **Appropriate Abstractions**

Different abstractions

(same or different formalisms)





- For performance (scale-ability)
- For **insight**

Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

TOWARDS ADAPTIVE ABSTRACTION IN AGENT BASED SIMULATION

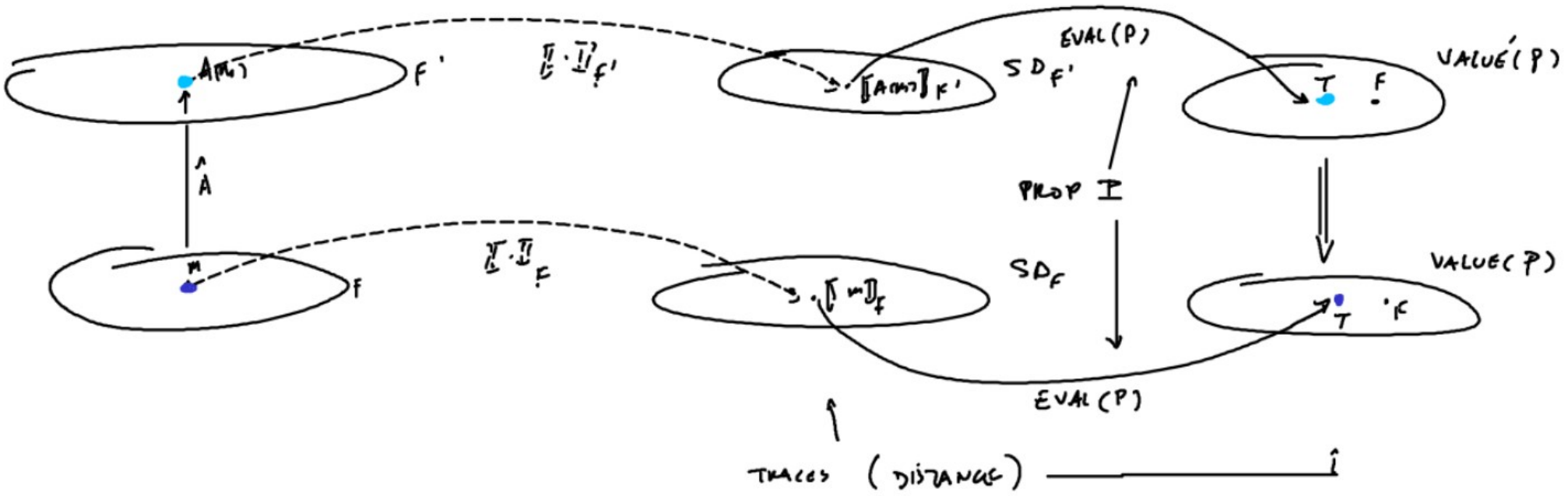
Romain Franceschini

University of Corsica Pasquale Paoli
UMR CNRS 6134
Campus Grimaldi
Corte, 20250, FRANCE

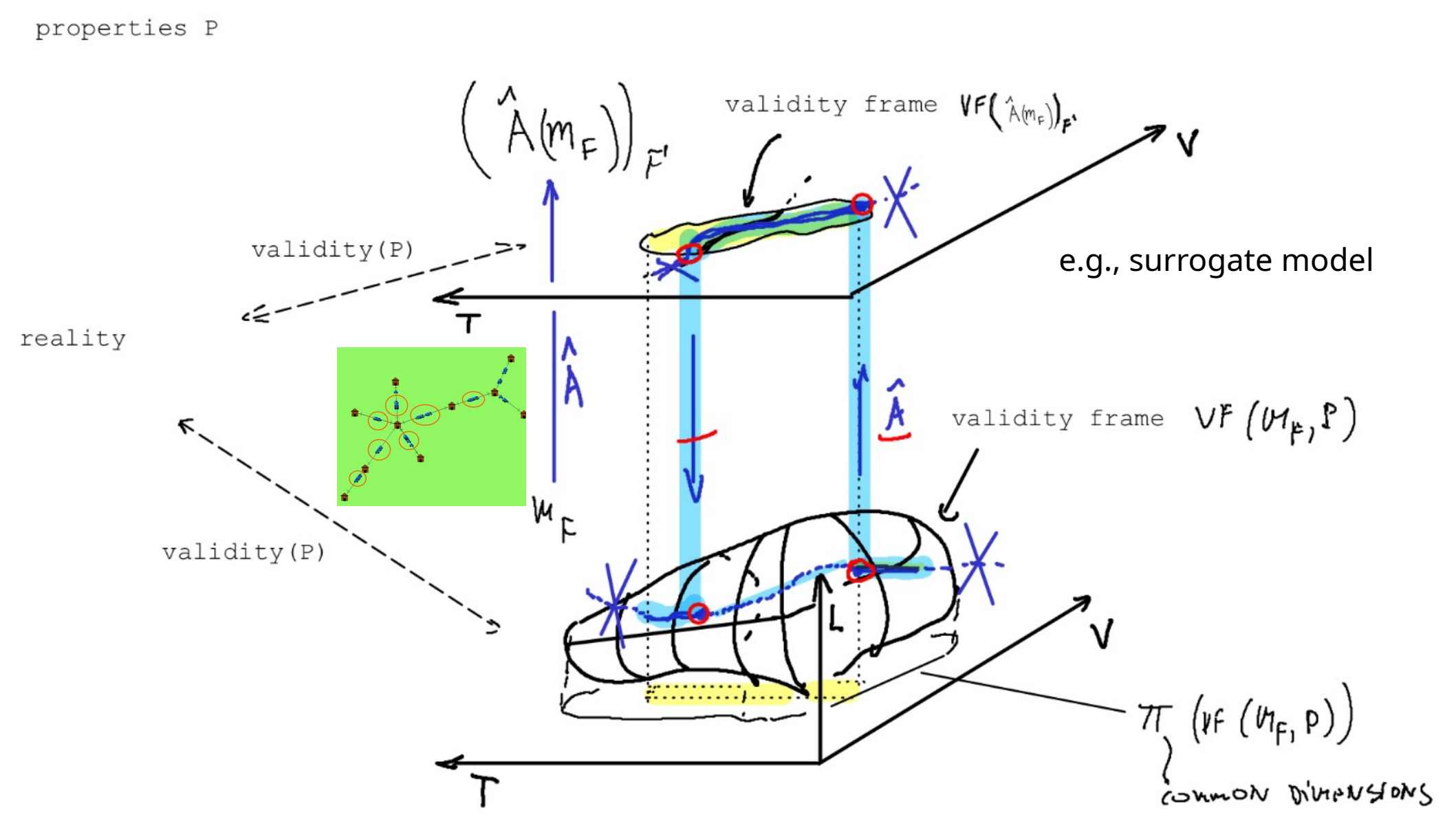
Simon Van Mierlo
Hans Vangheluwe

Department of Mathematics and Computer Science
University of Antwerp - Flanders Make
Middelheimlaan 1
Antwerp, 2020, BELGIUM

high performance

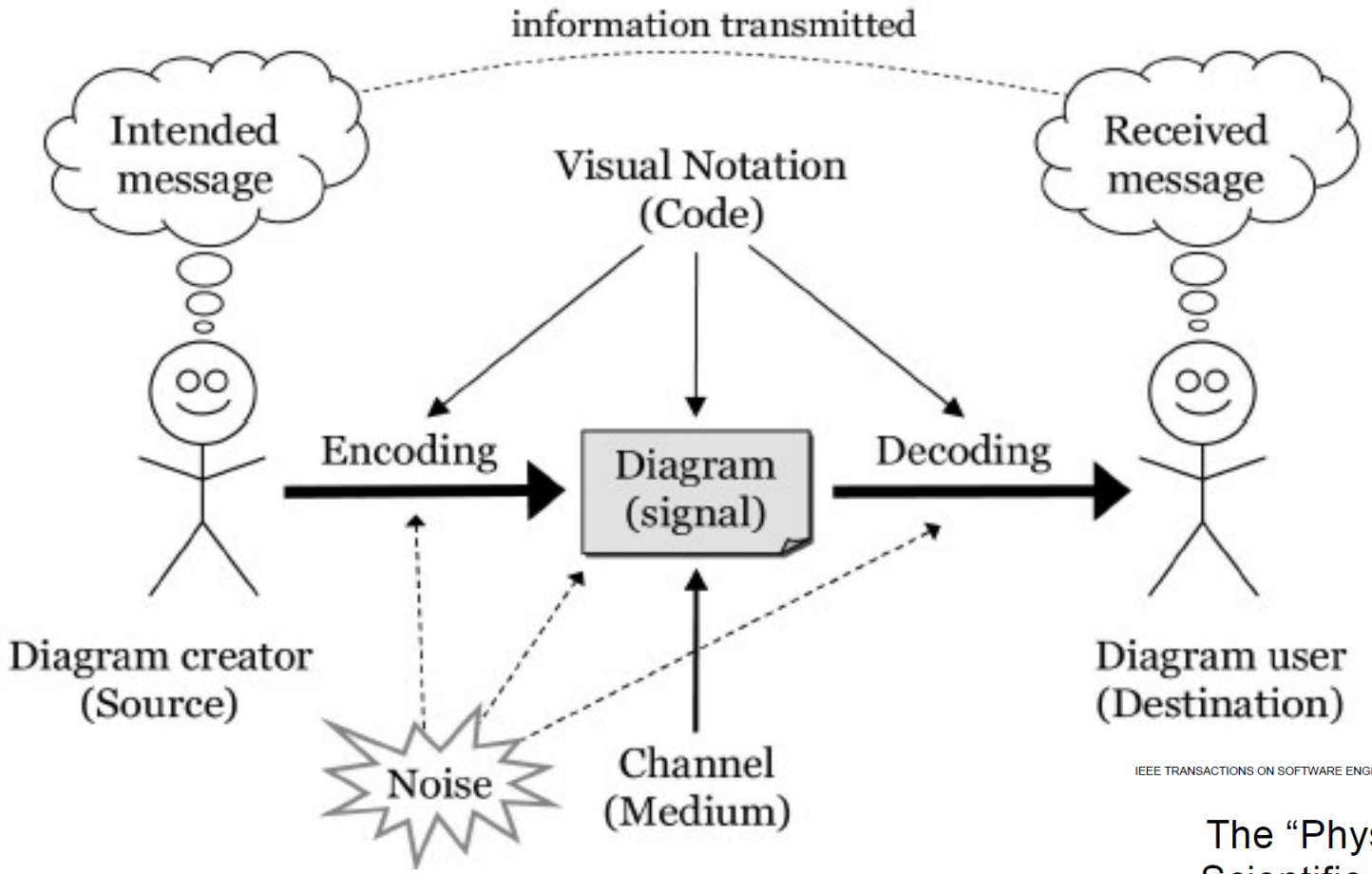


low performance


properties P
validity frame
validity(P)
reality
validity(P)
validity frame
e.g., surrogate model
common dimensions

Most **Appropriate Notations**

Communication Theory



IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 35, NO. 5, NOVEMBER-DECEMBER 2009

The “Physics” of Notations: Towards a Scientific Basis for Constructing Visual Notations in Software Engineering

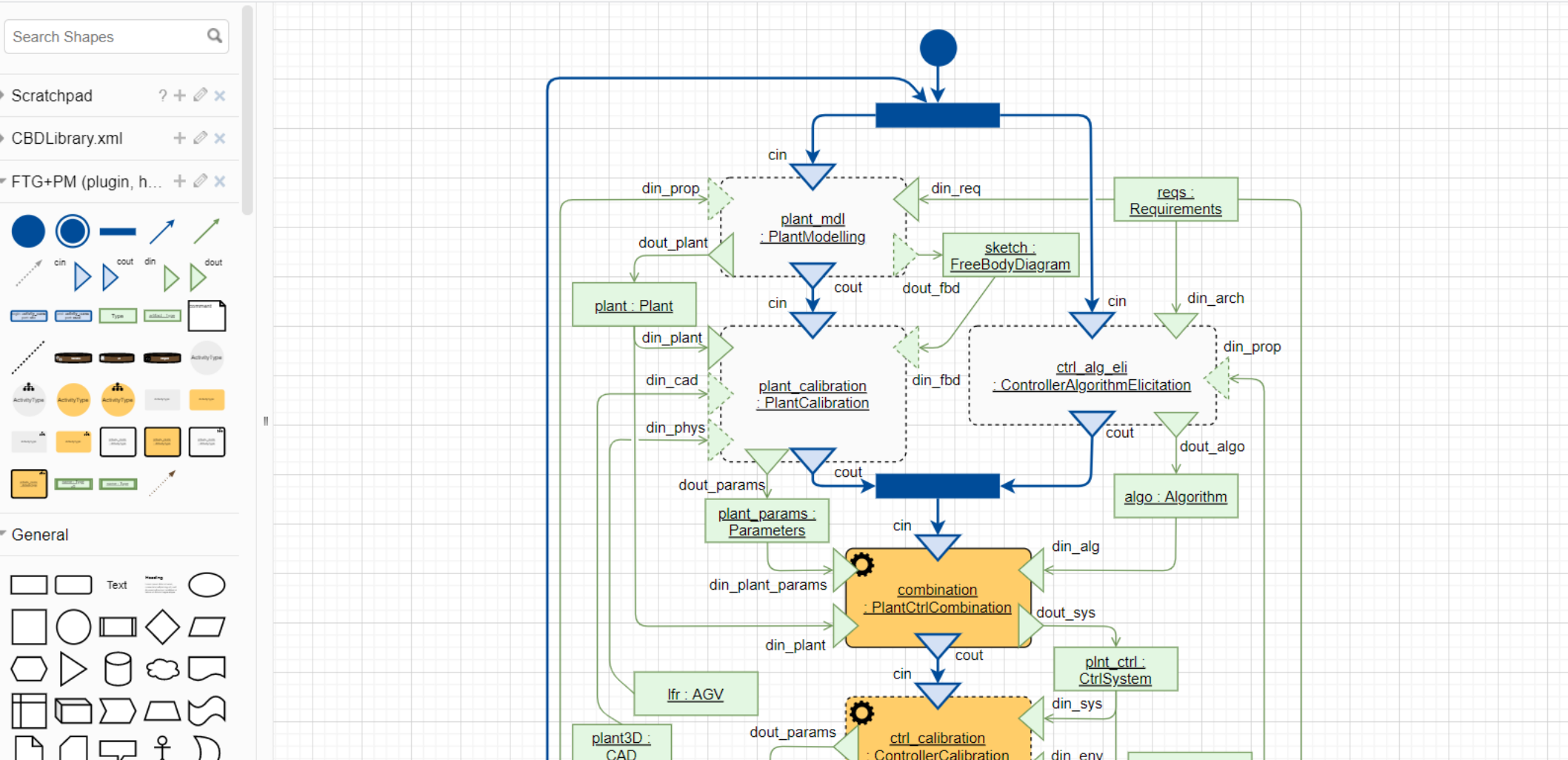
Daniel L. Moody, *Member, IEEE*

Semantic Transparency: semantically **perverse** symbols

FTGPMpp-2022-07-05.drawio

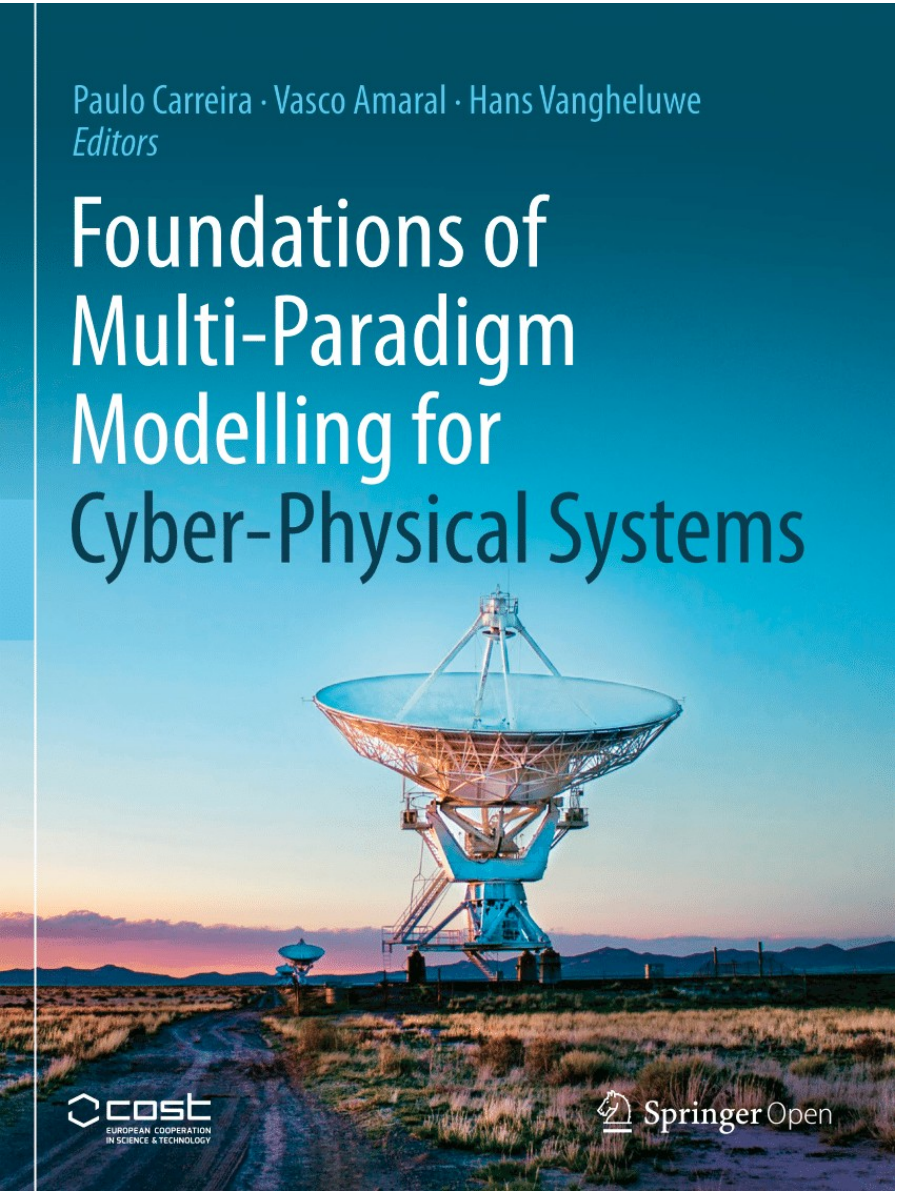
File Edit View Arrange Extras Help

https://diagrams.net (to build a “poor man’s” DSL)



Most **Appropriate Formalisms**

syntax **and** semantics



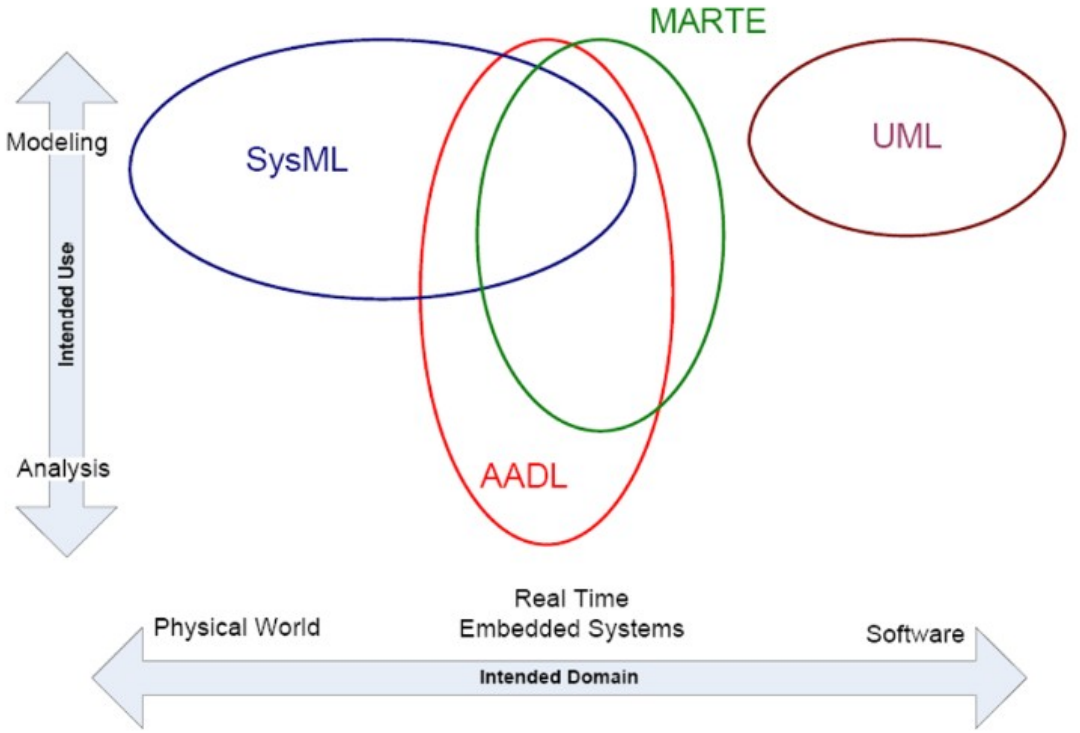
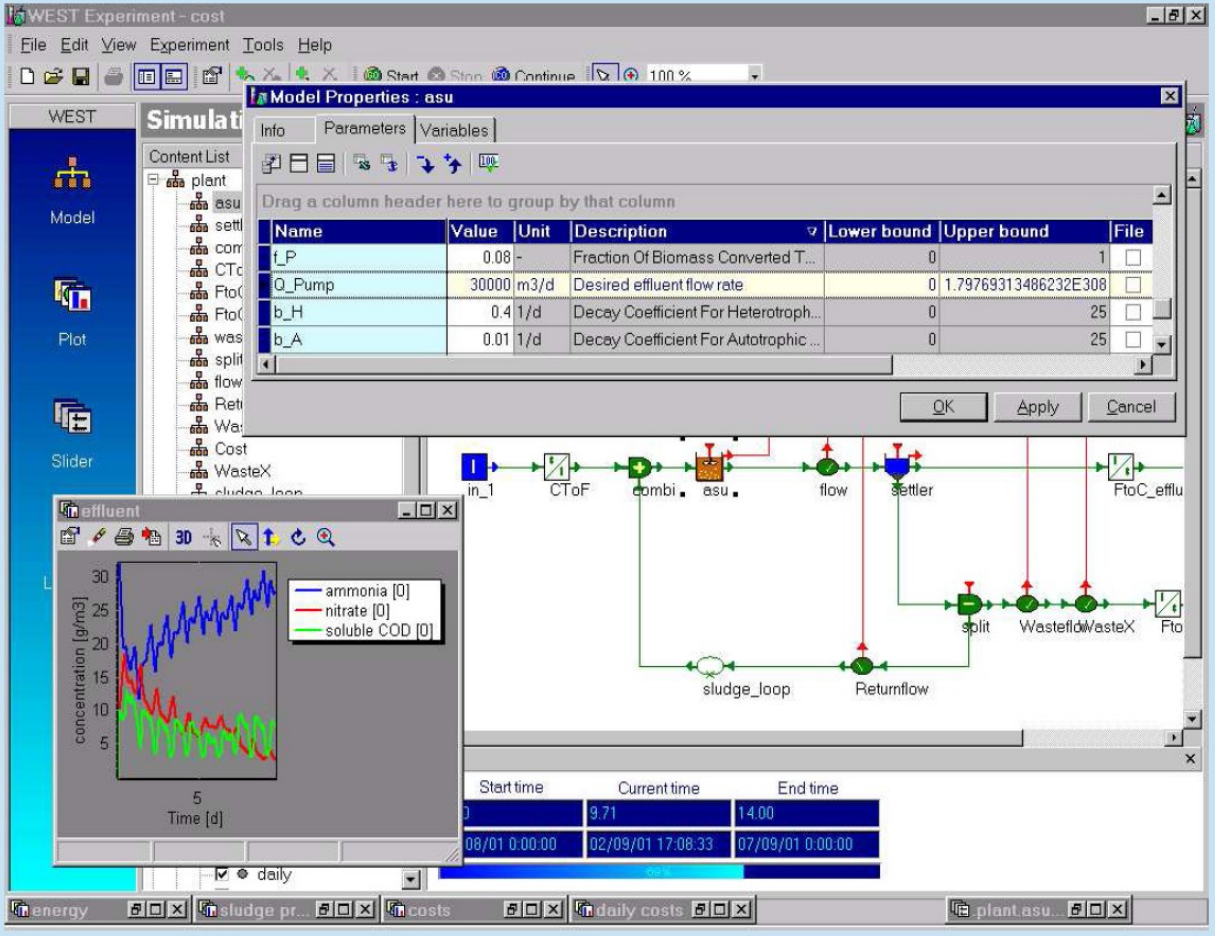


Fig. 8.5: Comparison of well-known ADLs in terms of intended use and domains (reproduced from [3])

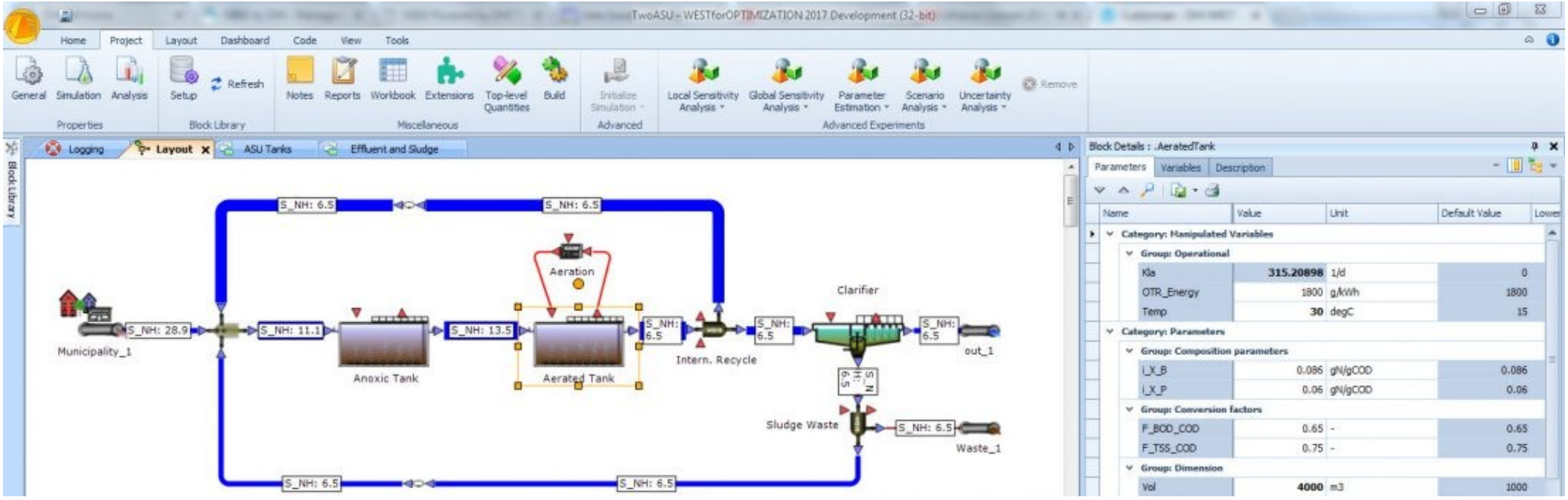
Carreira P., Amaral V., Vangheluwe H. (eds)
Foundations of Multi-Paradigm Modelling for Cyber-Physical Systems. Springer.

https://doi.org/10.1007/978-3-030-43946-0_2

DS(V)M Environment

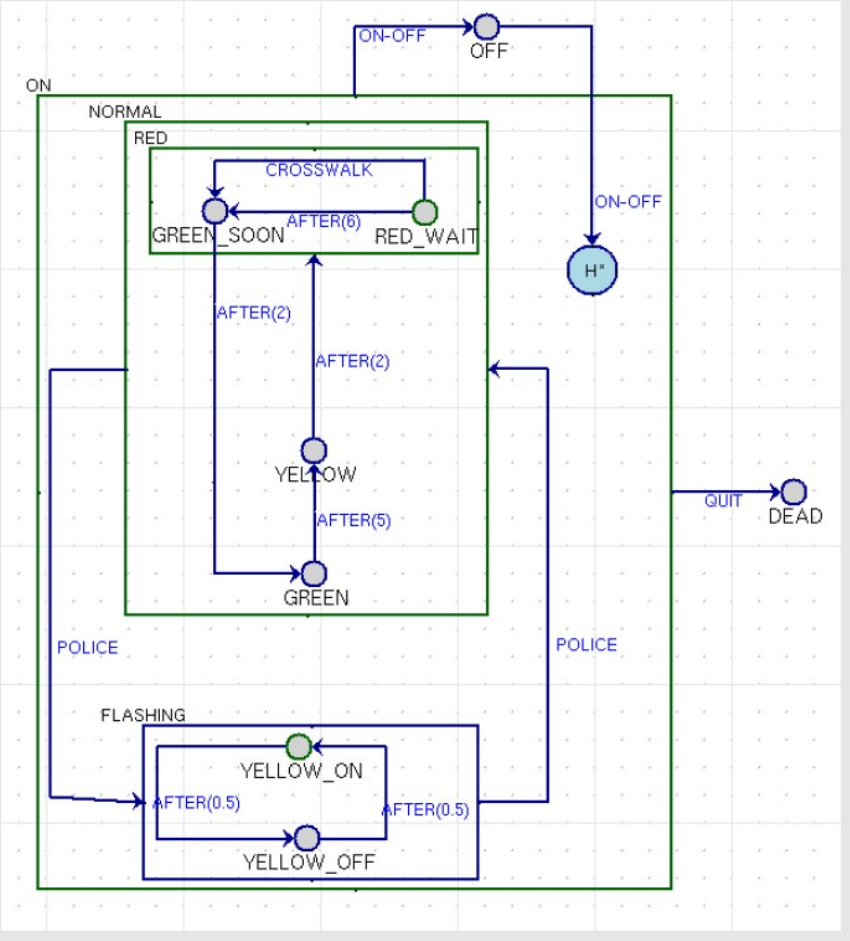


WEST: modelling biological wastewater treatment.
Henk Vanhooren, Jurgen Meirlaen, Youri Amerlinck, Filip Claeys, Hans Vangheluwe and Peter A.Vanrolleghem. Journal of Hydroinformatics 5 (2003) 27-50



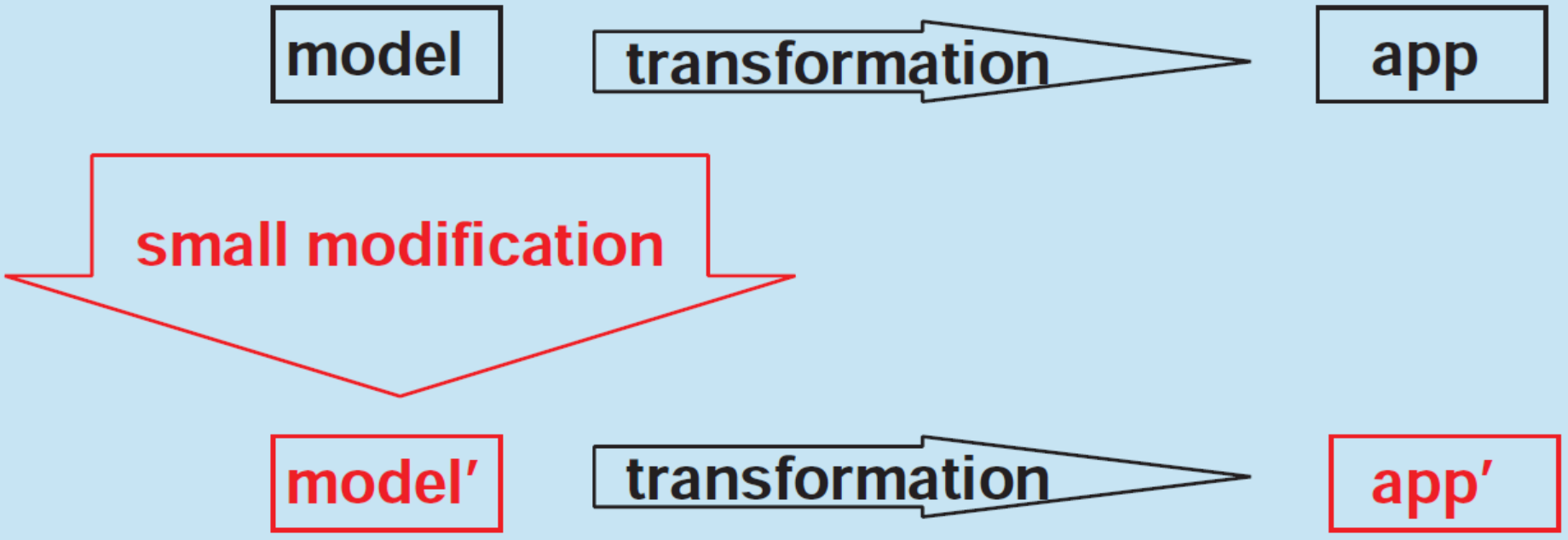
http://www.mikebydhi.com/products/west

Most Appropriate Formalism



Metrics?

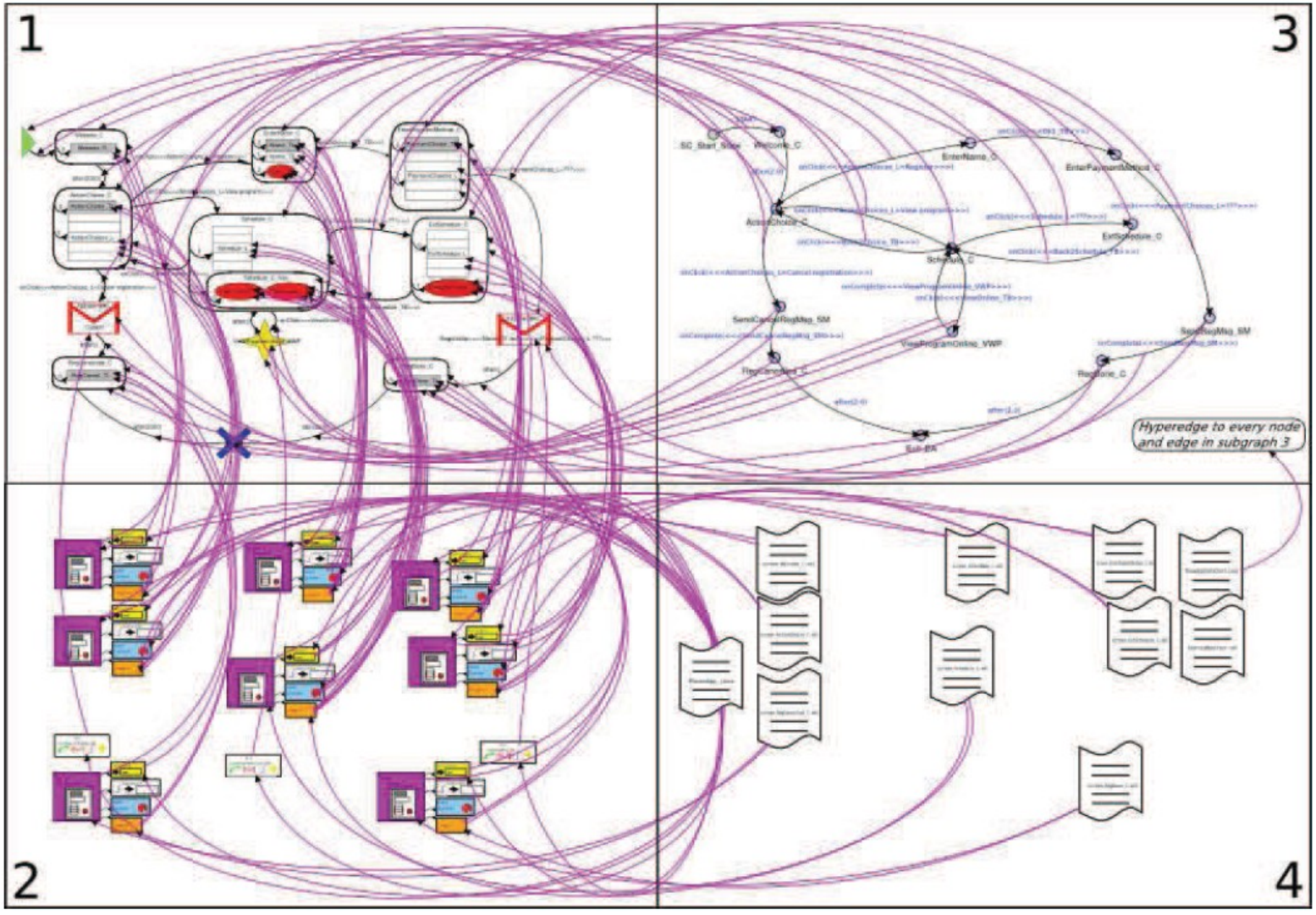
Model-Based Development:
Modify the Model
(e.g., based on feature model of product family)



small **modification** in model may lead to large change in **app**

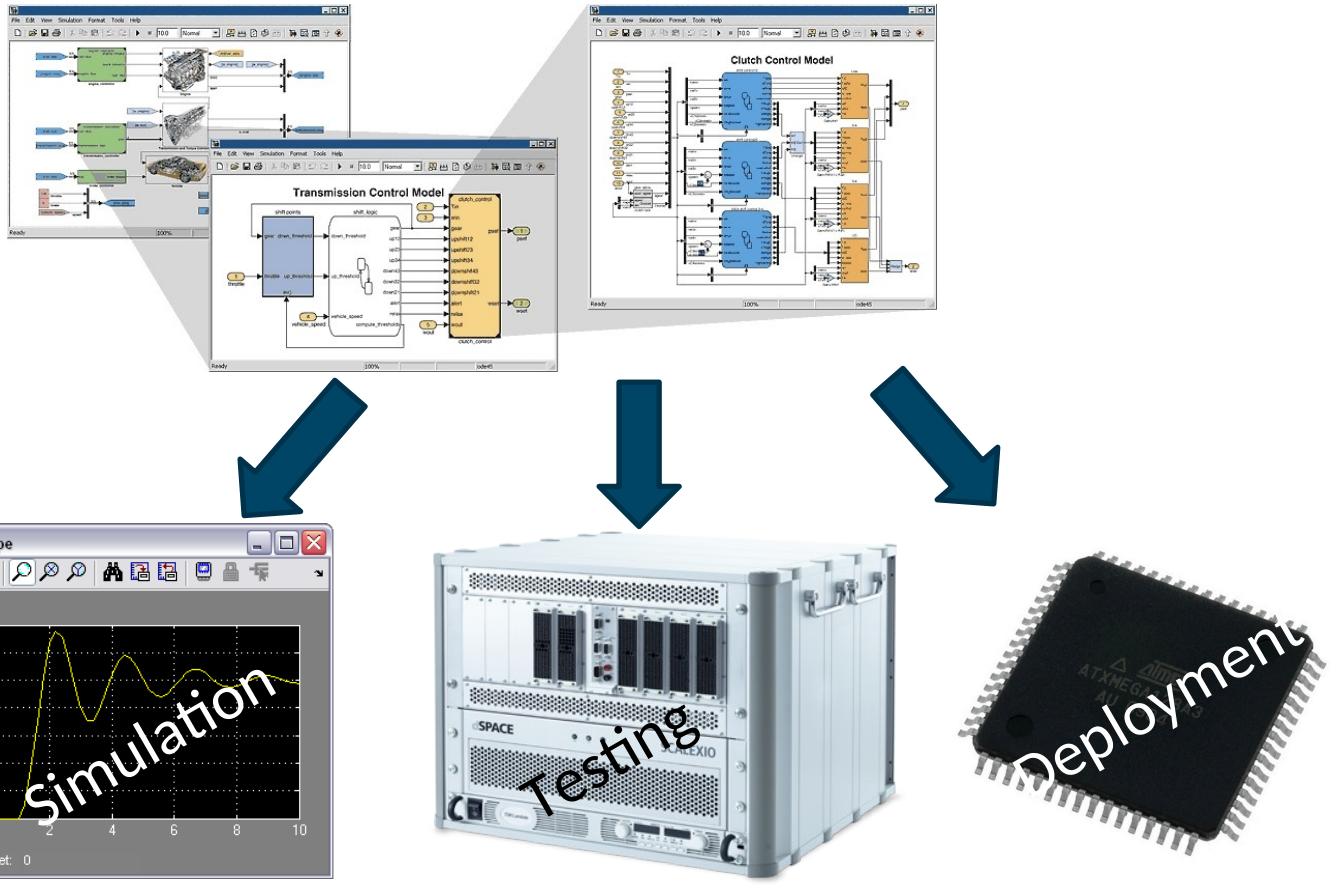
$\sim$ choice of formalism (e.g., Statecharts)

Can be Multi-Step/Multi-Formalism

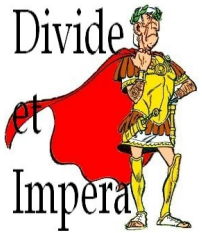


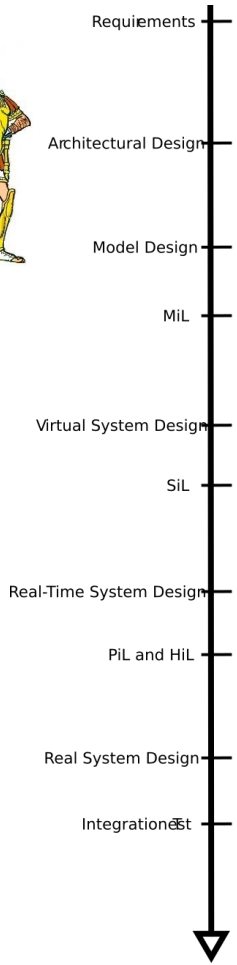
most appropriate level of **deployment**

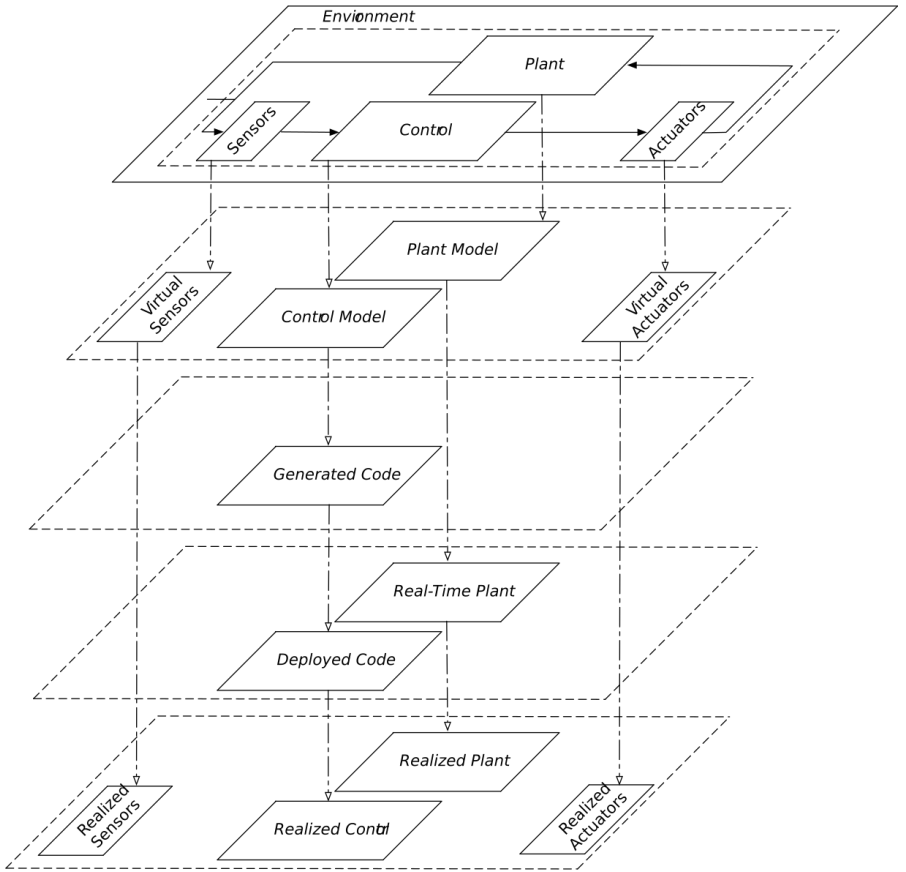
Model-Based System Design



XiL: X = Model, Software, Processor, Hardware







vertical consistency!

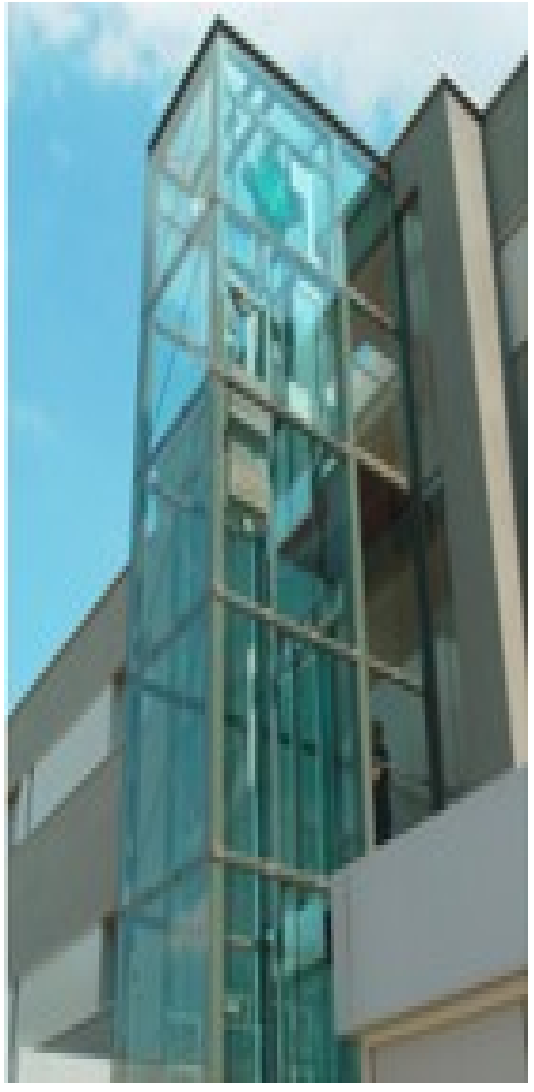
DT Valley

EPSim

Ken Vanherpen. A contract-based approach for multi-viewpoint consistency in the concurrent design of cyber-physical systems. PhD thesis University of Antwerp. 2018.

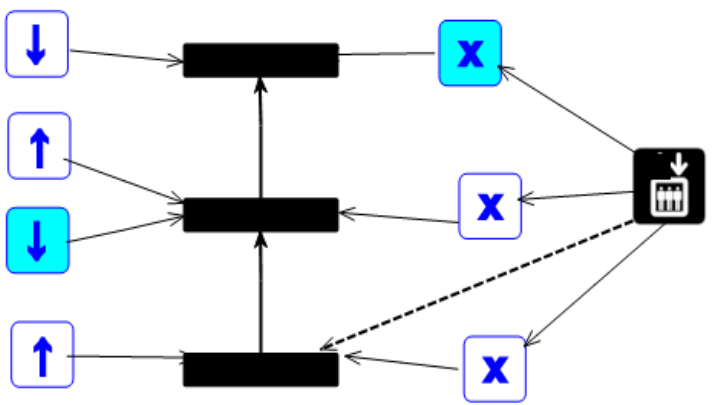
kinds of models that **always belong together**

”ProMoBox”



Designing Requirements/Property Languages

Design ⊨ Property

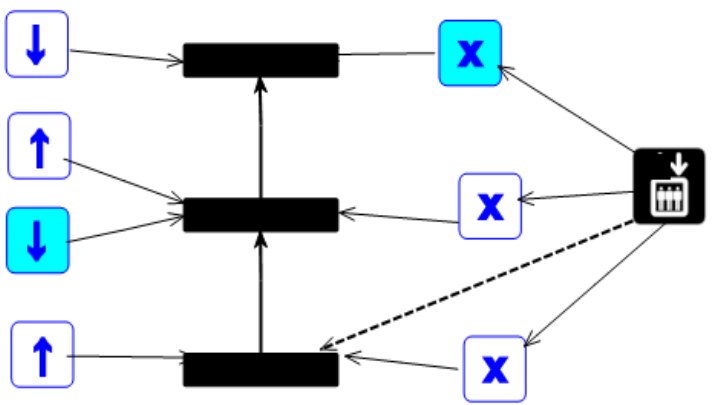


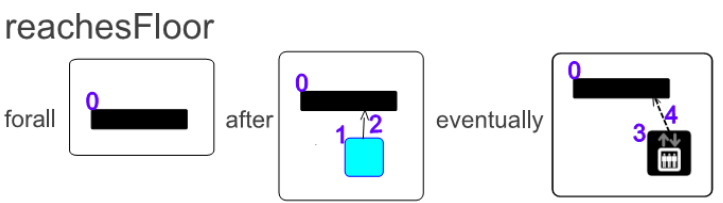
$$\Box(((go0 \wedge up0) \vee \Diamond(floor0 \vee idle)) \rightarrow ((\neg(floor0) \vee \neg(floor0 \vee idle))\mathcal{U}((floor0 \vee idle) \wedge (((floor0) \vee \neg(floor0 \vee idle))\mathcal{U}((floor0 \vee idle) \wedge ((\neg(floor0) \vee \neg(floor0 \vee idle))\mathcal{U}((floor0 \vee idle) \wedge (((floor0) \vee \neg(floor0 \vee idle))\mathcal{U}((floor0 \vee idle) \wedge (\neg(floor0)\mathcal{U}(floor0 \vee idle))))))))))) \vee \Box(((go1 \wedge up1 \wedge down1) \vee \Diamond(floor1 \vee idle)) \rightarrow ((\neg(floor1) \vee \neg(floor1 \vee idle))\mathcal{U}((floor1 \vee idle) \wedge (((floor1) \vee \neg(floor1 \vee idle))\mathcal{U}((floor1 \vee idle) \wedge ((\neg(floor1) \vee \neg(floor1 \vee idle))\mathcal{U}((floor1 \vee idle) \wedge (((floor1) \vee \neg(floor1 \vee idle))\mathcal{U}((floor1 \vee idle) \wedge (\neg(floor1)\mathcal{U}(floor1 \vee idle))))))))))) \vee \Box(((go2 \wedge down2) \vee \Diamond(floor2 \vee idle)) \rightarrow ((\neg(floor2) \vee \neg(floor2 \vee idle))\mathcal{U}((floor2 \vee idle) \wedge (((floor2) \vee \neg(floor2 \vee idle))\mathcal{U}((floor2 \vee idle) \wedge ((\neg(floor2) \vee \neg(floor2 \vee idle))\mathcal{U}((floor2 \vee idle) \wedge (((floor2) \vee \neg(floor2 \vee idle))\mathcal{U}((floor2 \vee idle) \wedge (\neg(floor2)\mathcal{U}(floor2 \vee idle)))))))))))$$

B. Meyers, R. Deshayes, L. Lucio, E. Syriani, H. Vangheluwe, and M. Wimmer. ProMoBox: A Framework for Generating Domain-Specic Property Languages. In Software Language Engineering (SLE), Vasteras, Sweden, LNCS vol. 8706, pp. 1- 20. Springer. September 2014.

Designing Requirements/Property Languages

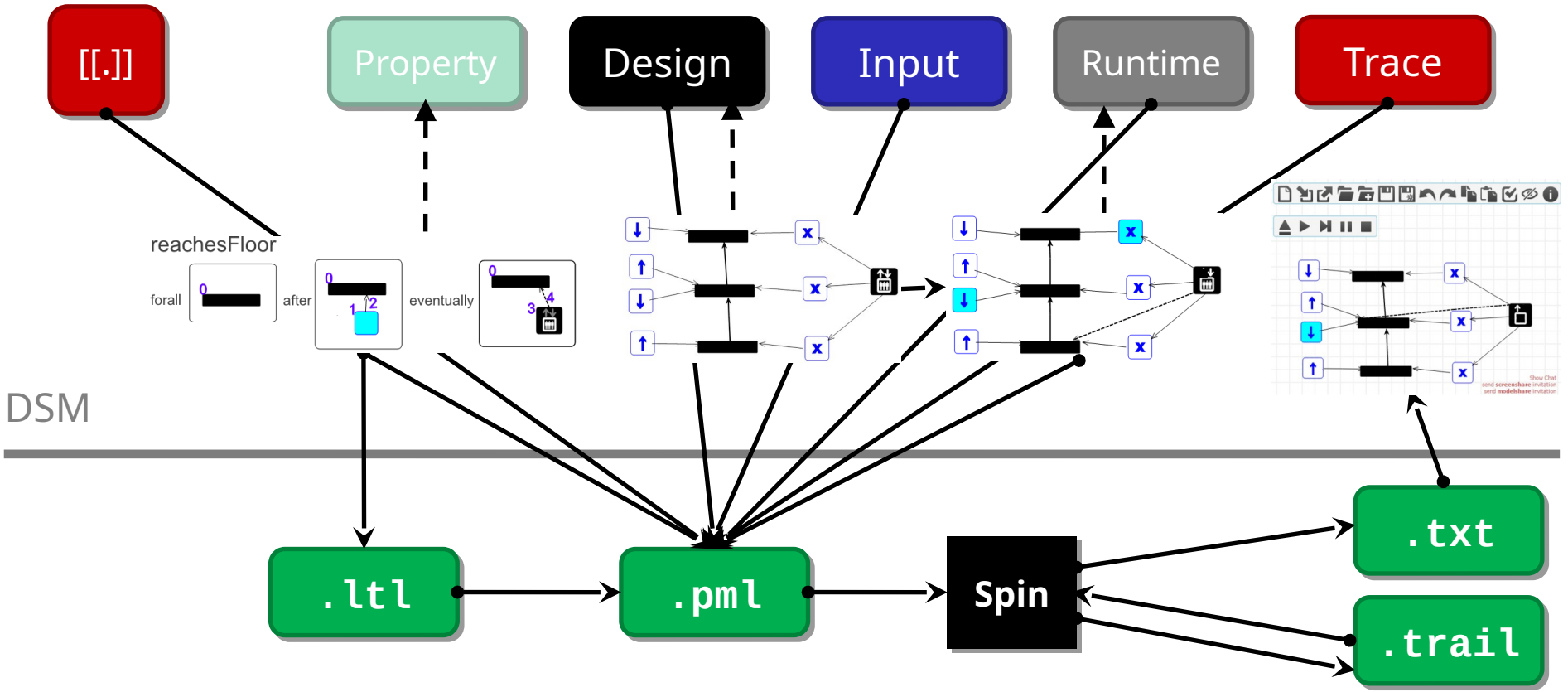
Design ⊨ Property





B. Meyers, R. Deshayes, L. Lucio, E. Syriani, H. Vangheluwe, and M. Wimmer. ProMoBox: A Framework for Generating Domain-Specic Property Languages. In Software Language Engineering (SLE), Vasteras, Sweden, LNCS vol. 8706, pp. 1- 20. Springer. September 2014.

Designing DS Requirements/Property Languages

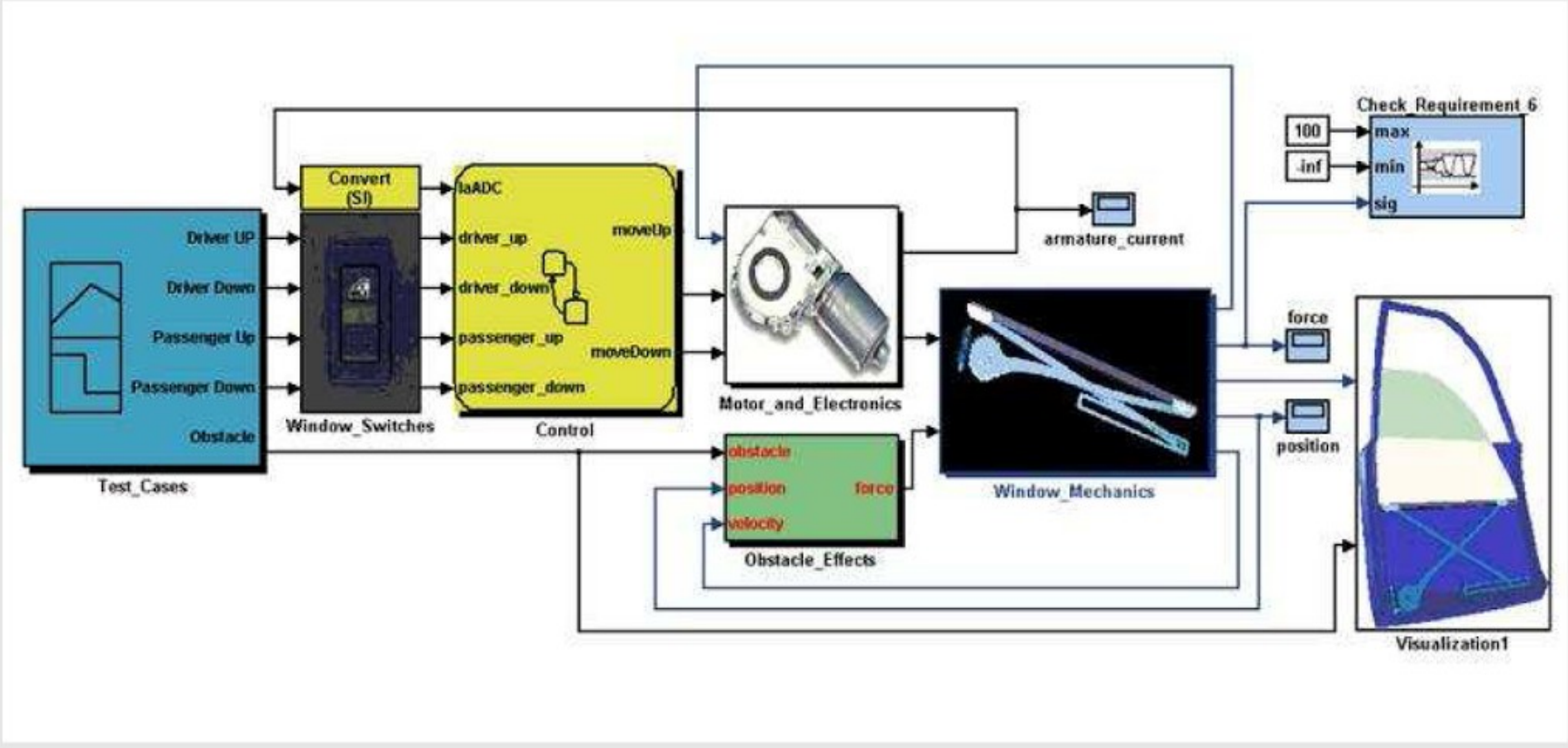


B. Meyers, H. Vangheluwe, J. Denil and R. Salay, "A Framework for Temporal Verification Support in Domain-Specific Modelling,"
in IEEE Transactions on Software Engineering. doi:10.1109/TSE.2018.2859946

Most Appropriate **Combination** of Formalisms:

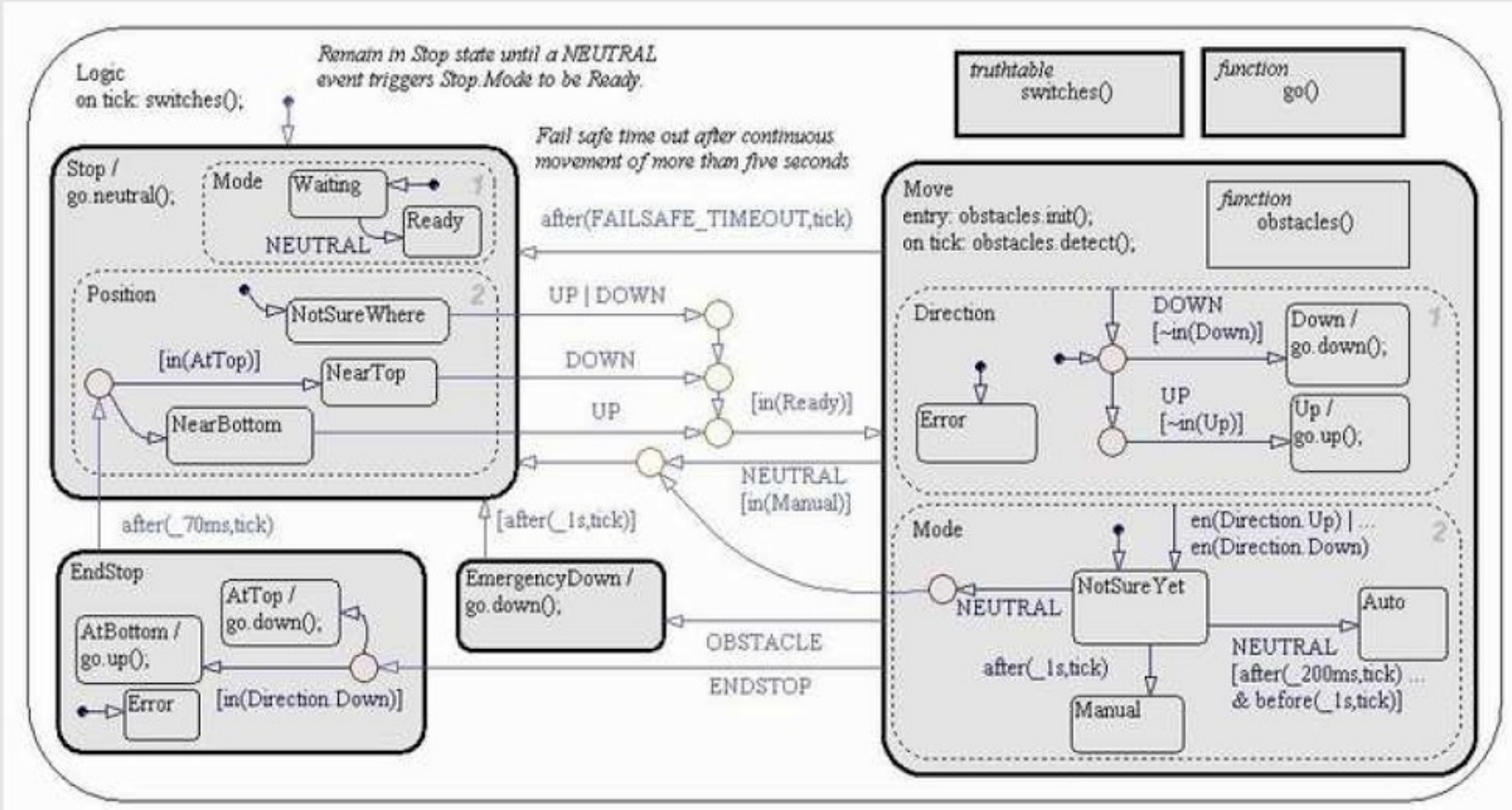
architectures

Components in Different Formalisms

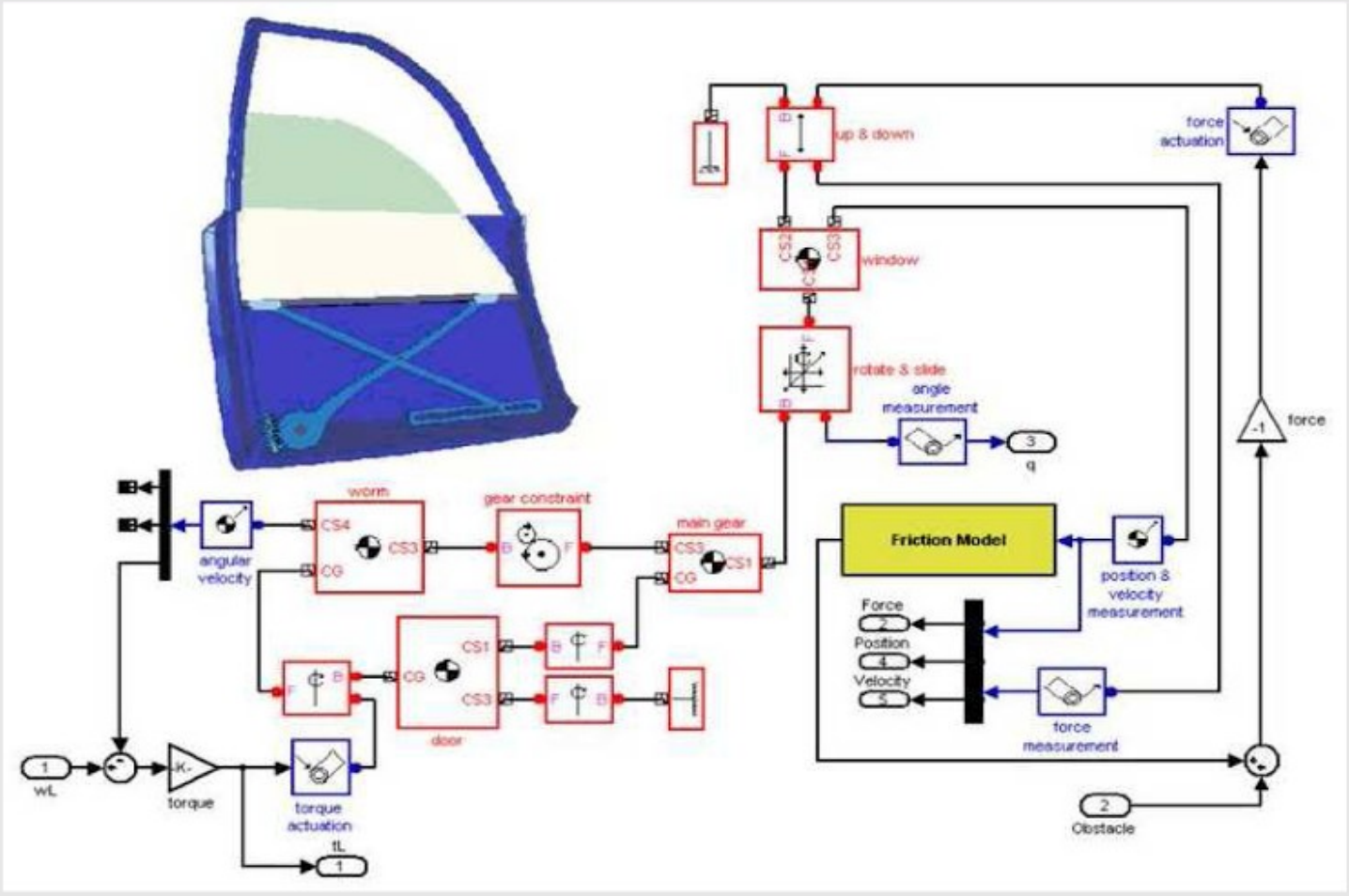


www.mathworks.com/products/demos/simulink/PowerWindow/html/PowerWindow1.html

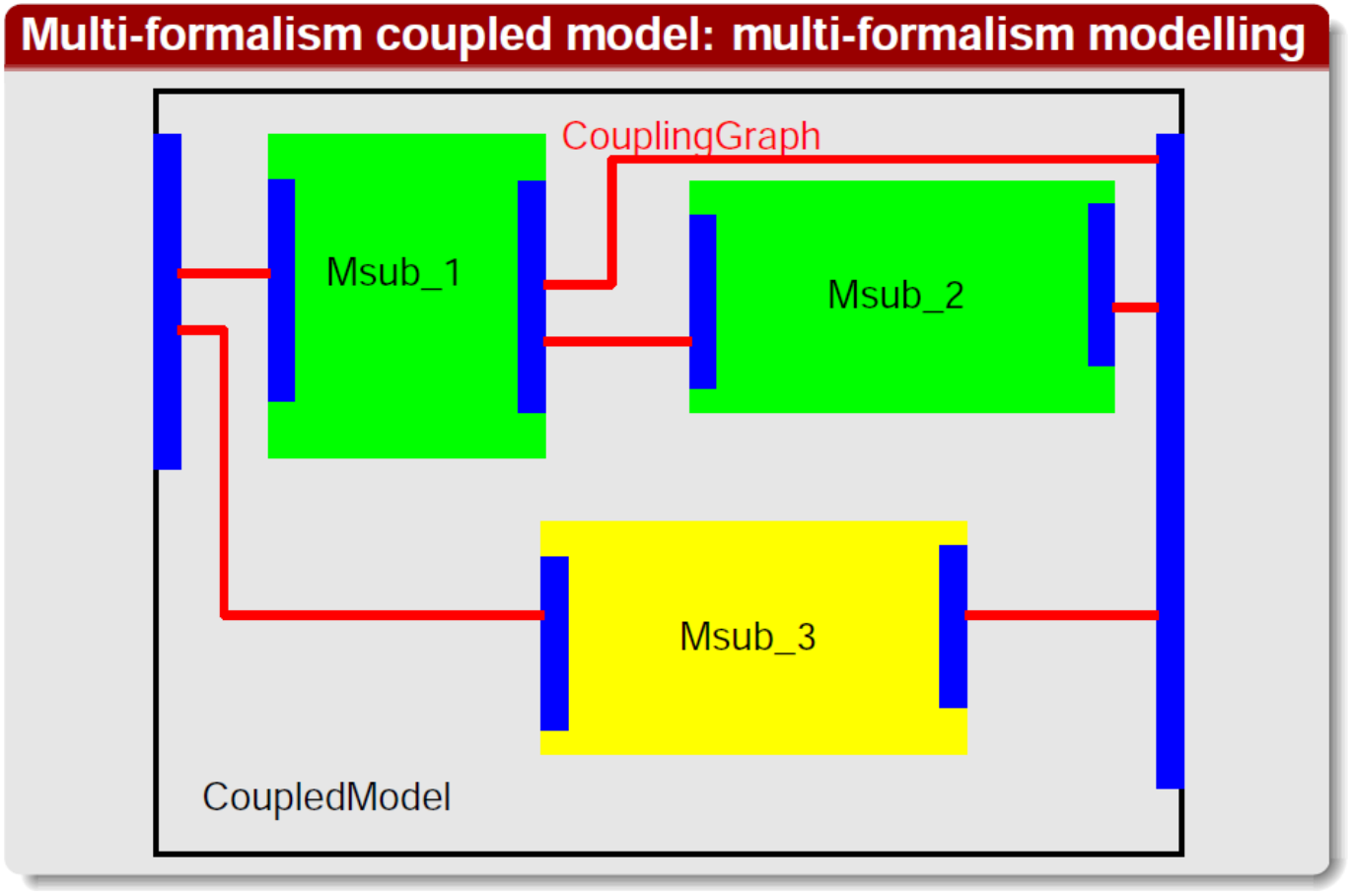
Controller, using Statechart(StateFlow) formalism



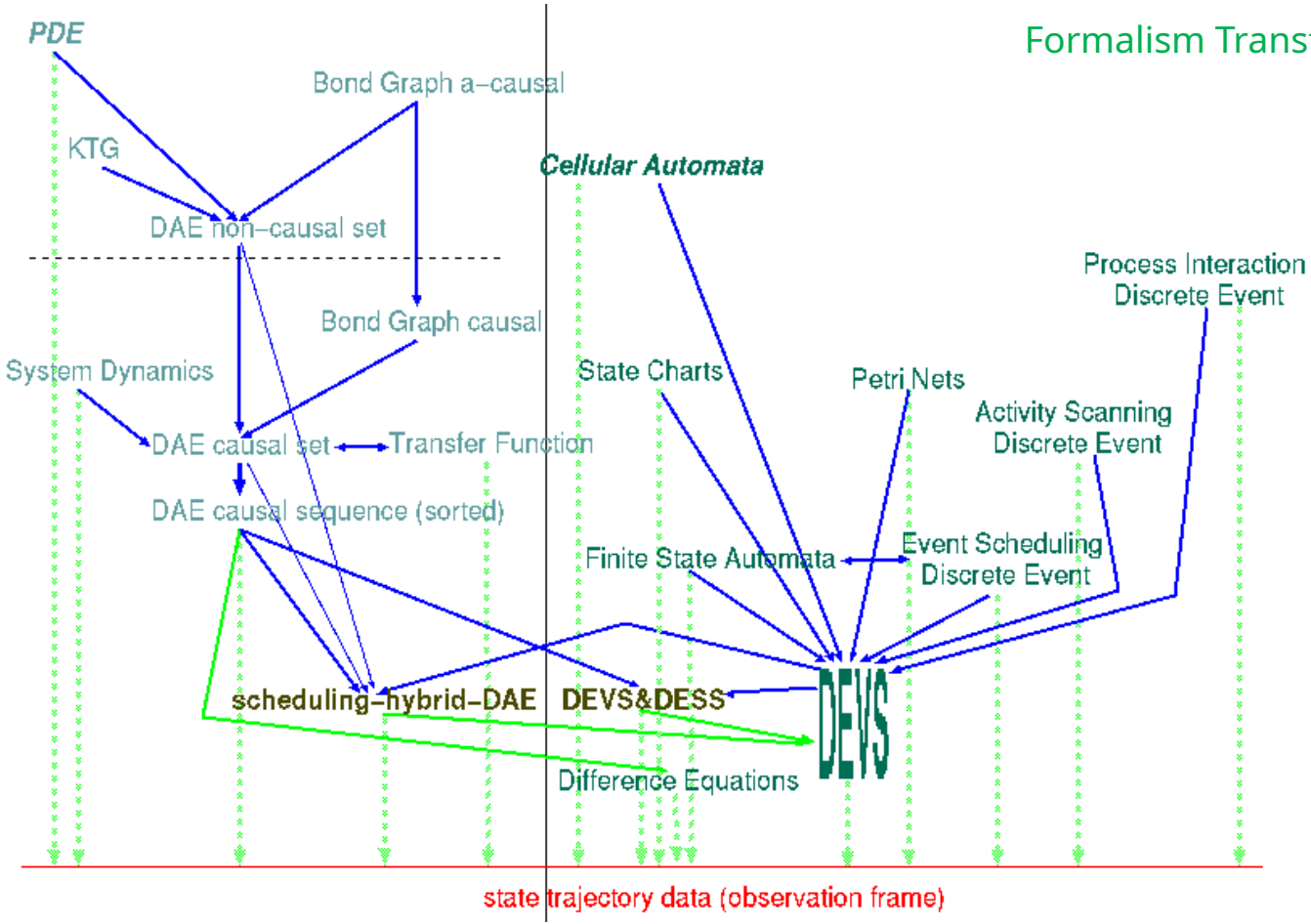
Mechanics subsystem



Bran Selić: “fragmentation problem”



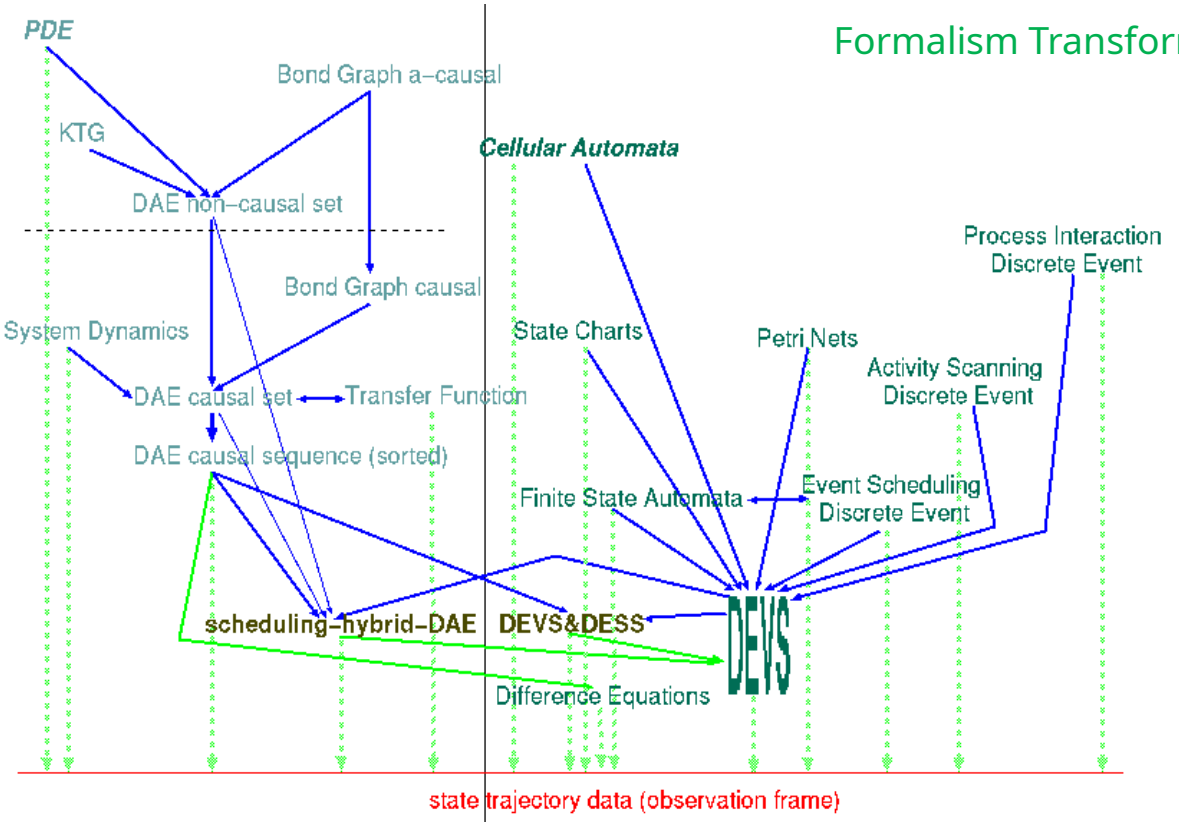
Formalism Transformation Graph (FTG)

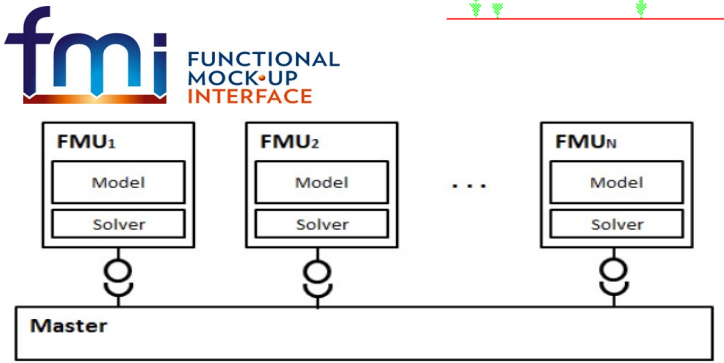


Caveat: proving semantics/property preservation of a single transformation (denoted by a blue arrow) may take at least one PhD thesis!

Hans Vangheluwe and Ghislain C. Vansteenkiste. A multi-paradigm modeling and simulation methodology: Formalisms and languages. In European Simulation Symposium (ESS) , pages 168 – 172. Society for Computer Simulation International (SCS), October 1996. Genoa, Italy.

Formalism Transformation Graph (FTG) co-simulation



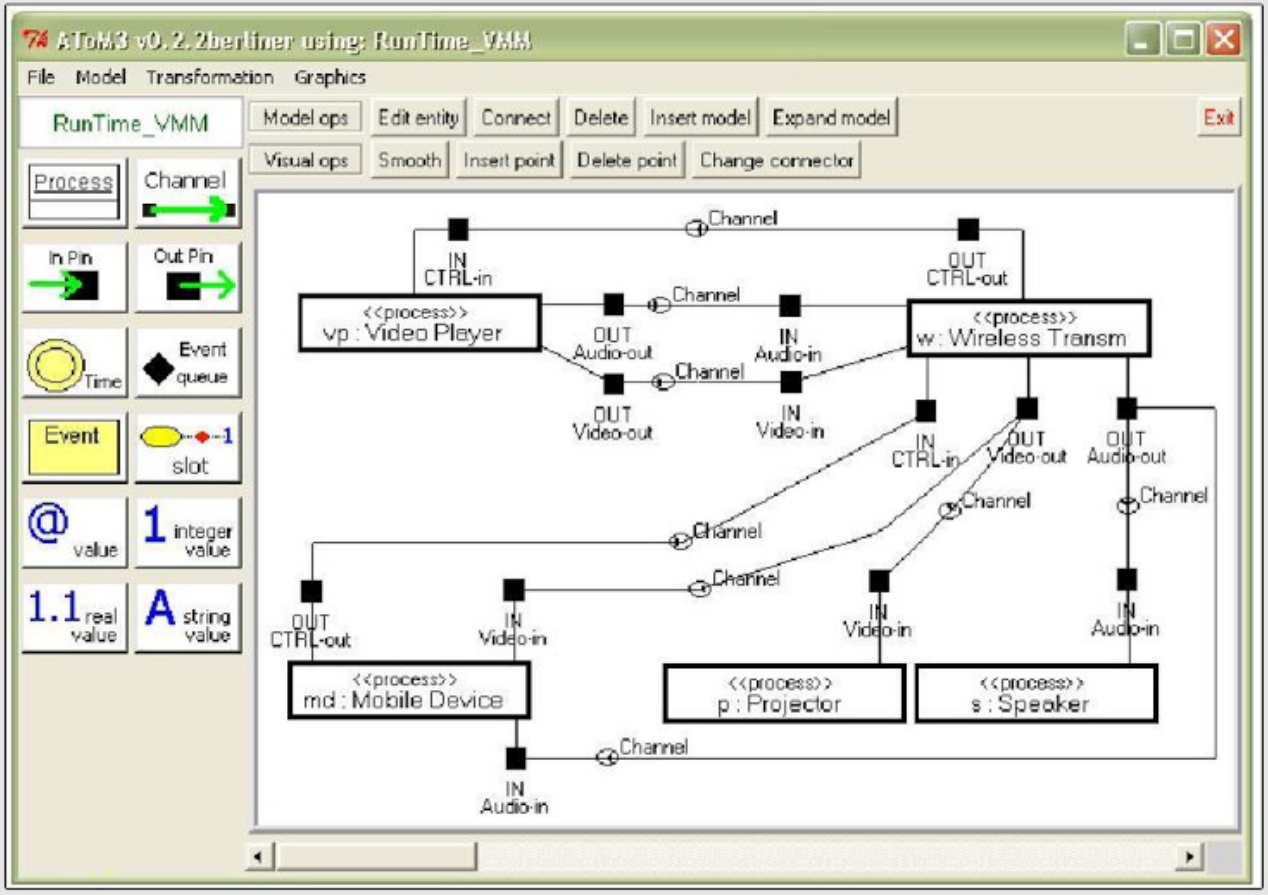


Cláudio Gomes, Casper Thule, David Broman, Peter Gorm Larsen, and Hans Vangheluwe. Co-simulation: A survey. ACM Computing Surveys (CSUR) , 51(3):49:1-49:33, 2018.

Most appropriate Views

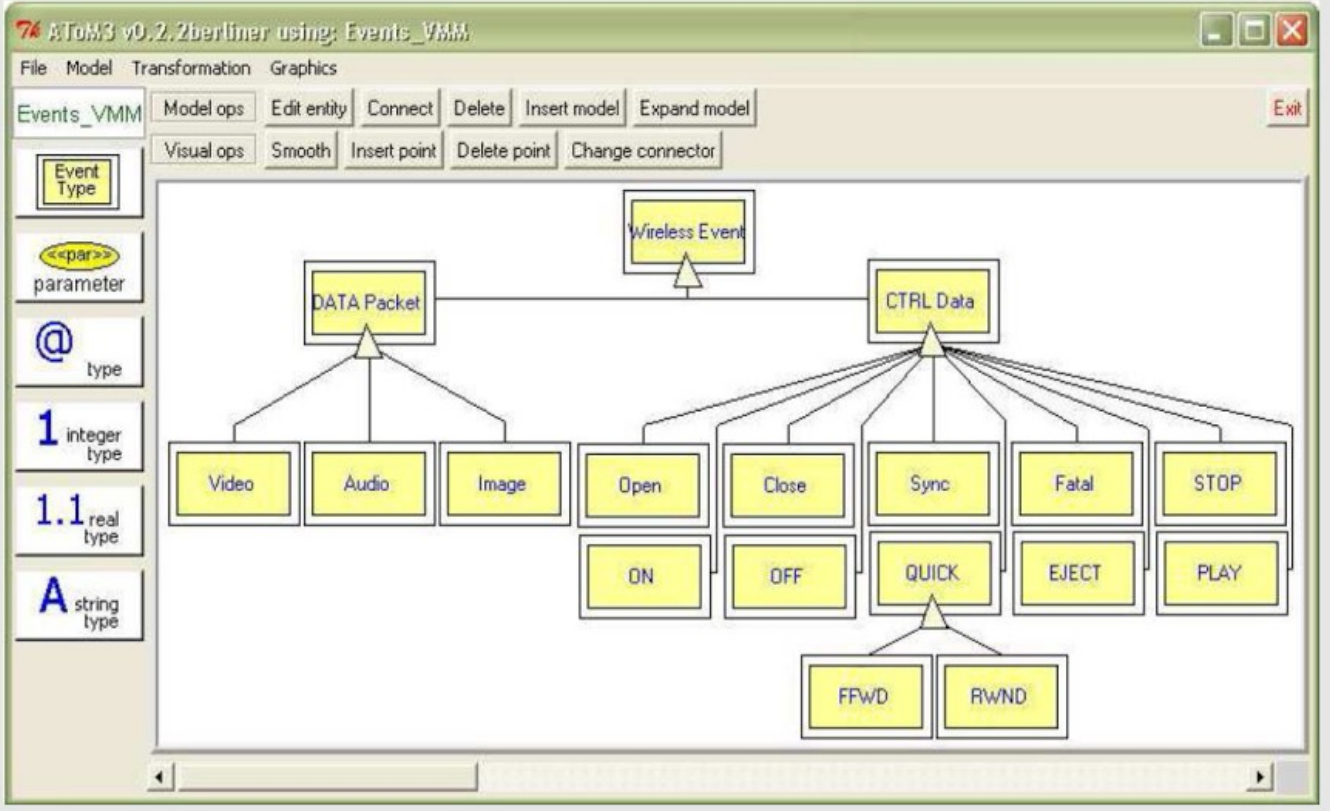
Wireless Home Entertainment System

Multiple (consistent !) Views (in ≠ Formalisms)

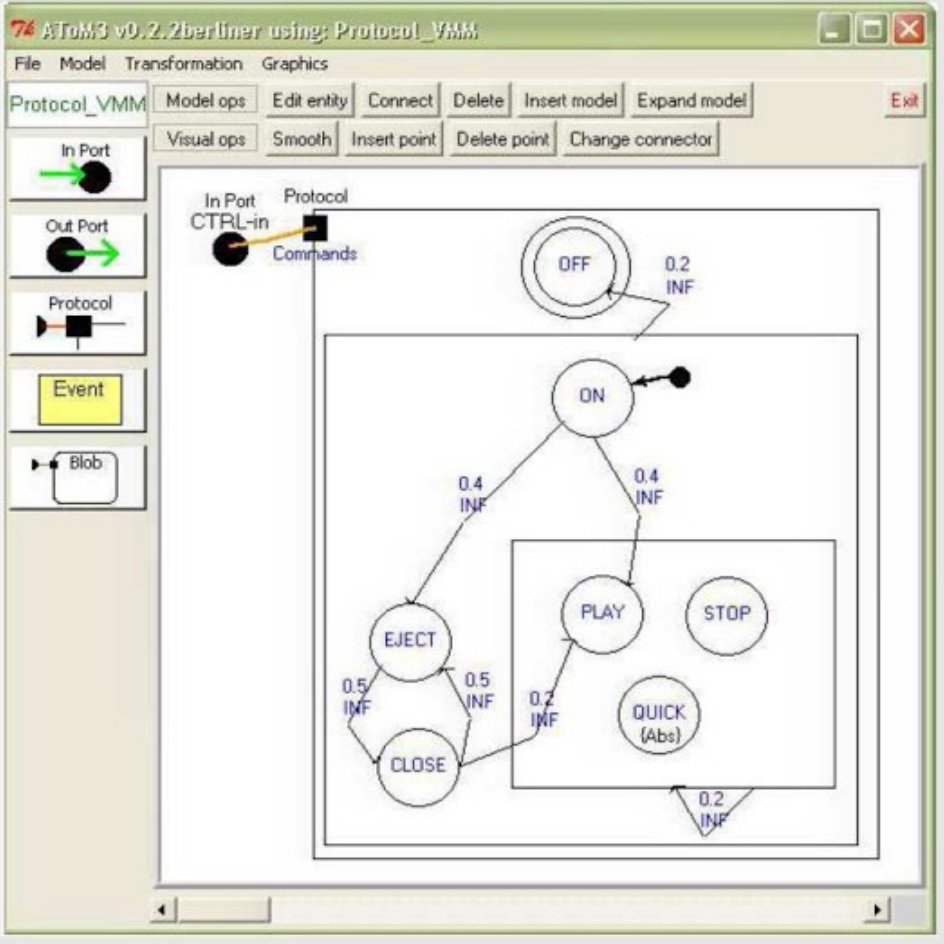


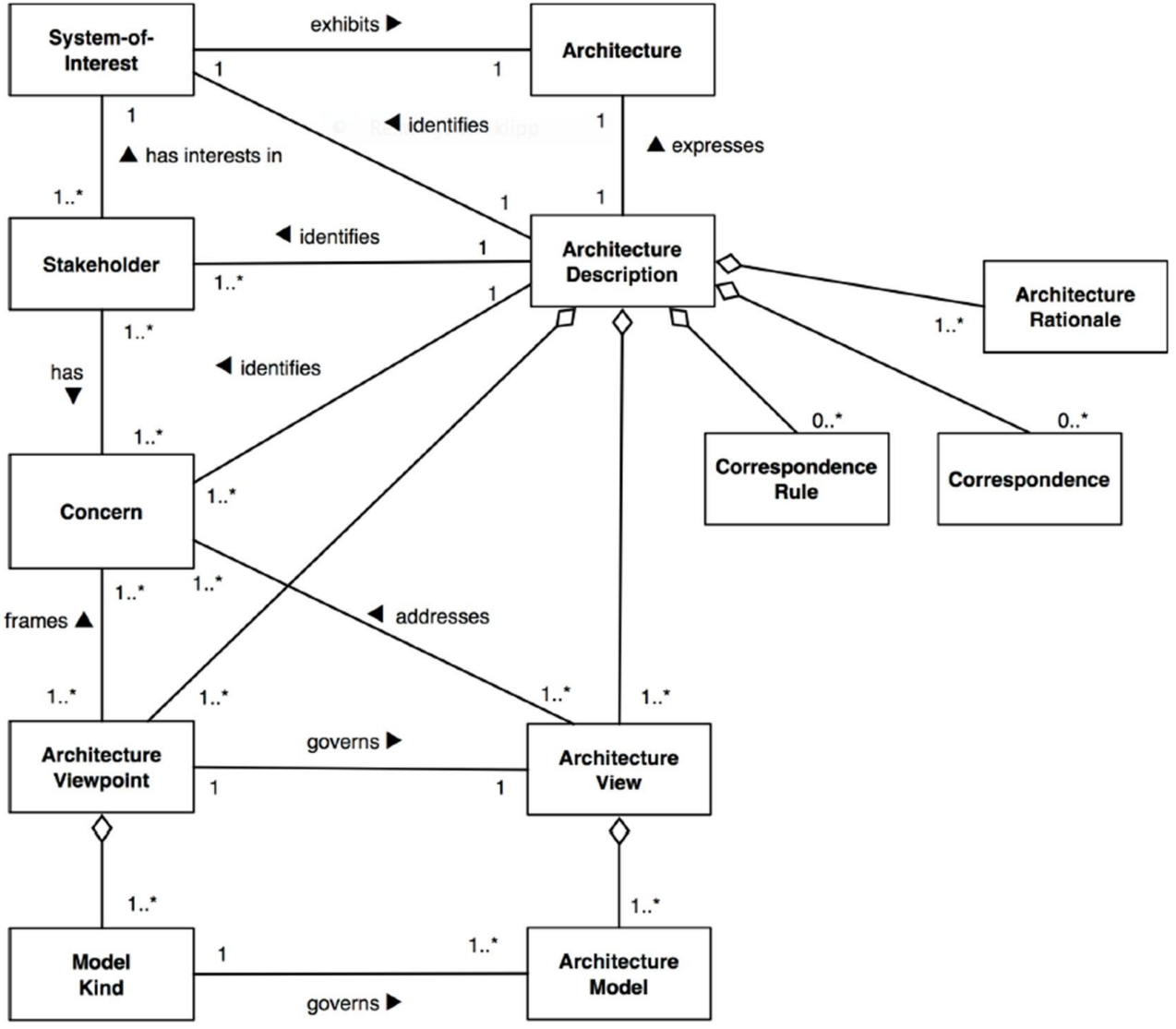
E. Guerra, P. Diaz and J. de Lara, A formal approach to the generation of visual language environments supporting multiple views. 2005 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'05), Dallas, TX, USA, 2005, pp. 284-286, doi: 10.1109/VLHCC.2005.6.

View: Events Diagram



View: Protocol Statechart





INTERNATIONAL STANDARD

ISO/IEC/ IEEE 42010

First edition
2011-12-01

Systems and software engineering — Architecture description

Journal of Computer Languages 76 (2023) 101223

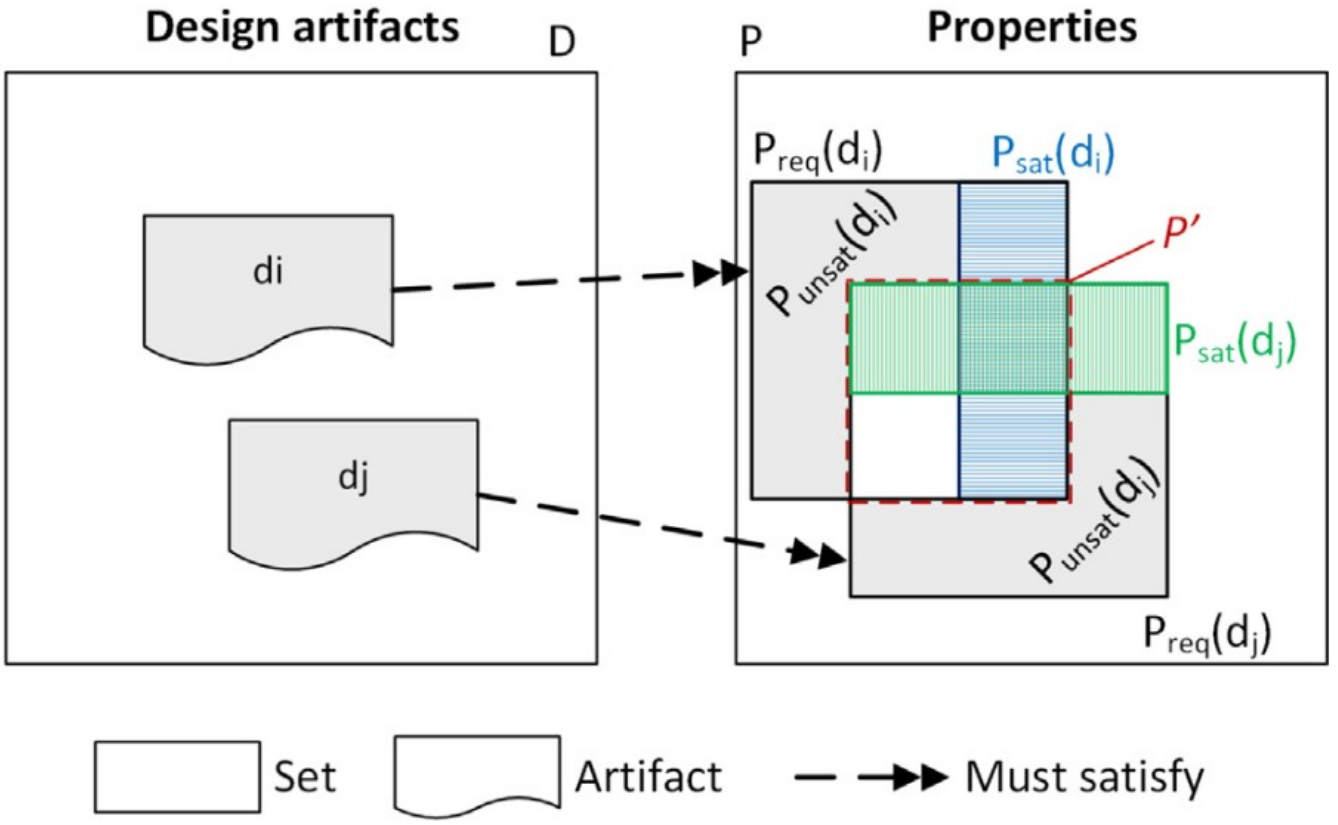
Contents lists available at ScienceDirect

Journal of Computer Languages

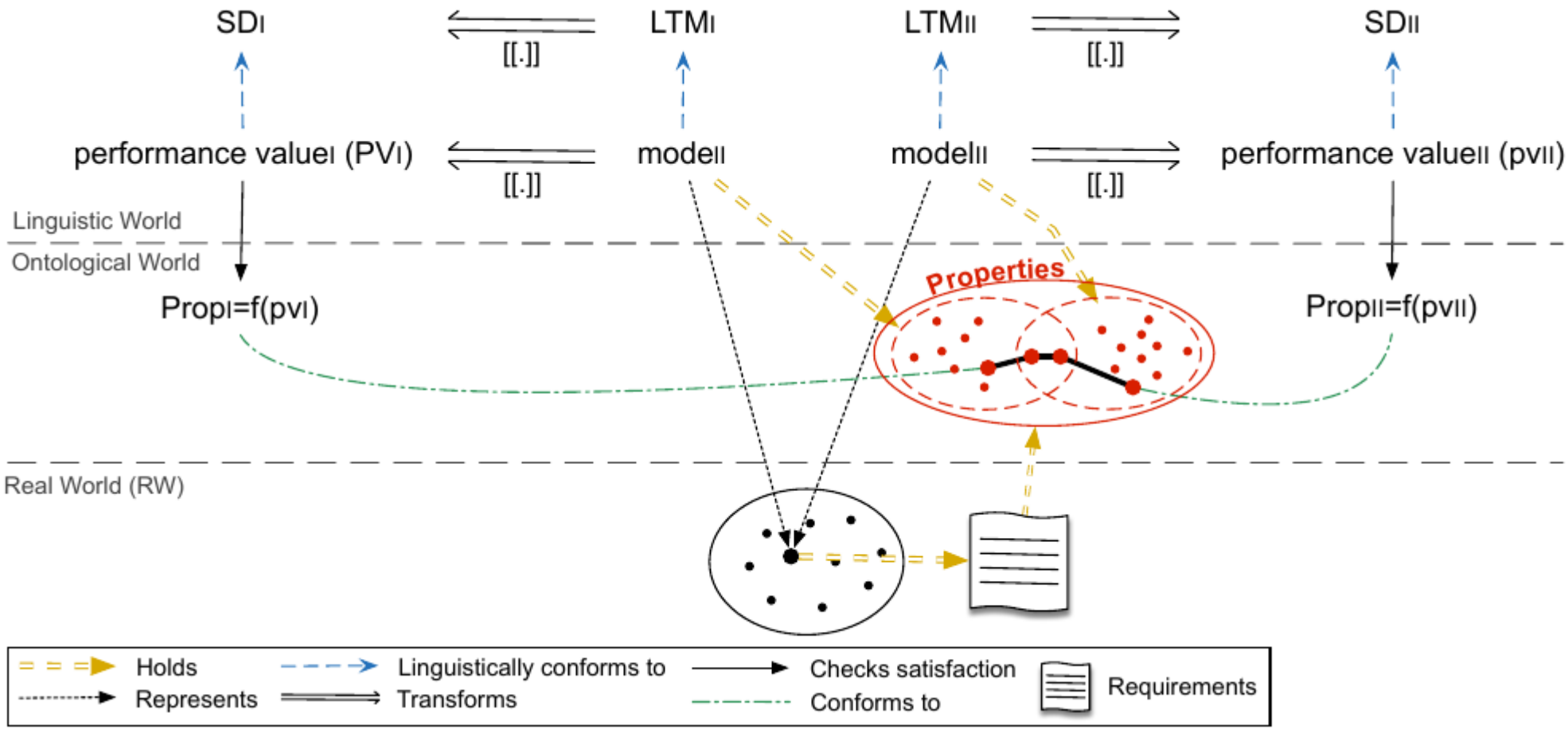
journal homepage: www.elsevier.com/locate/cola

Model consistency as a heuristic for eventual correctness

Istvan David [a,*], Hans Vangheluwe [b,c], Eugene Syriani [a]

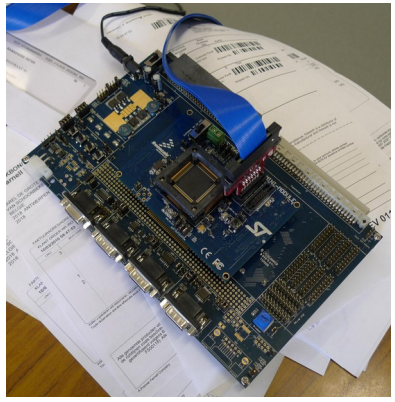


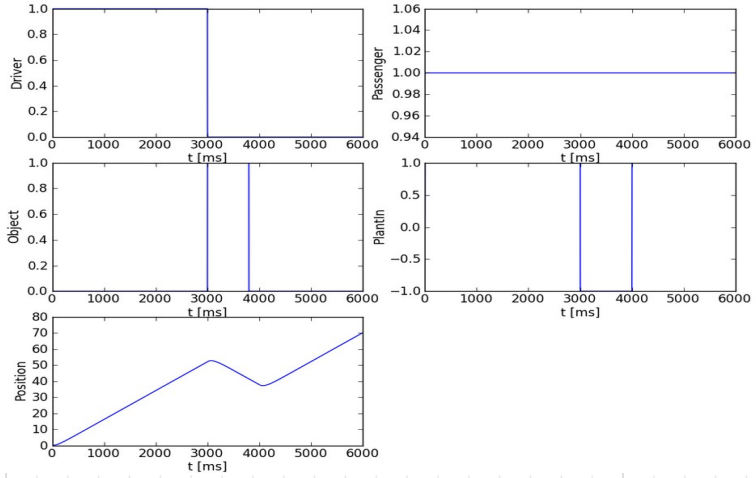
consistency across domains



Appropriate (and explicitly modelled) **Workflow**

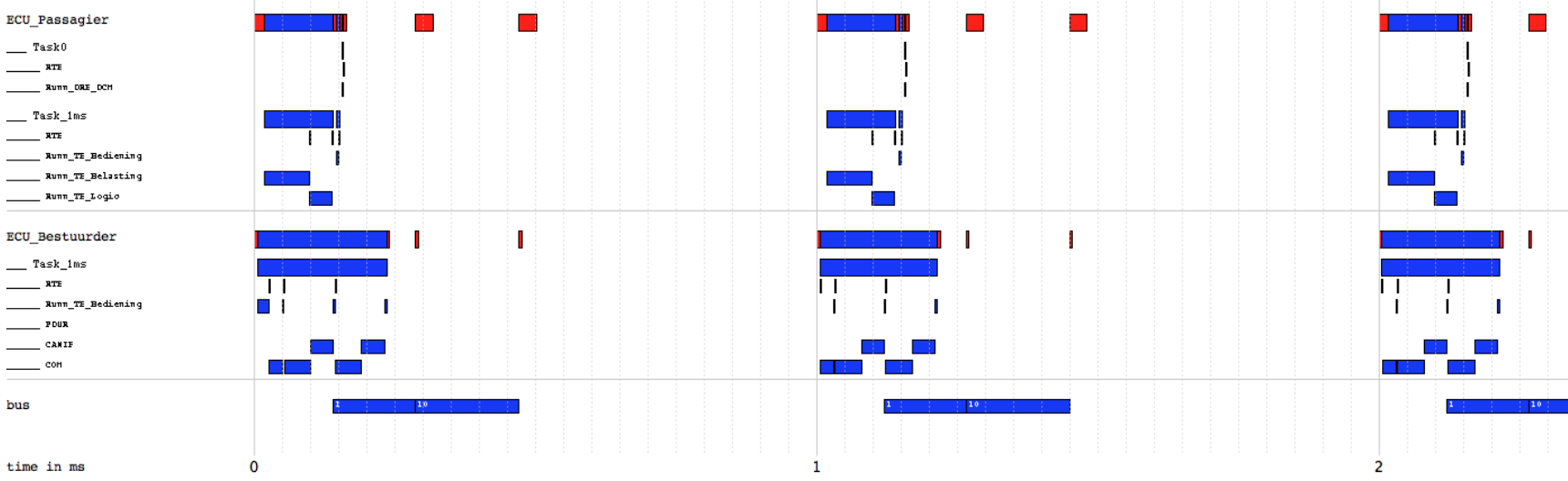
Deployment/Design-Space Exploration



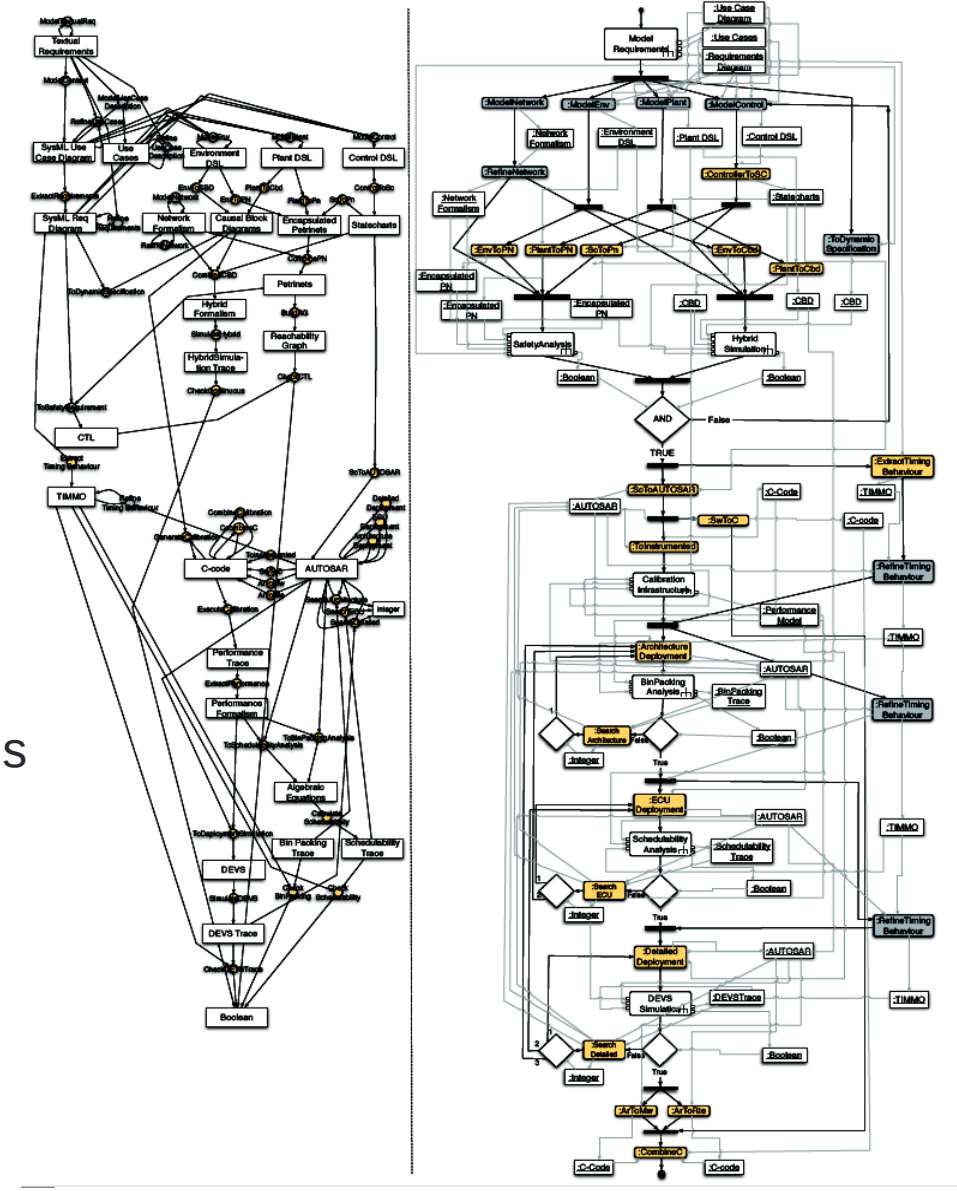


DT Valley

EPSim



FTG+PM (Process Model)

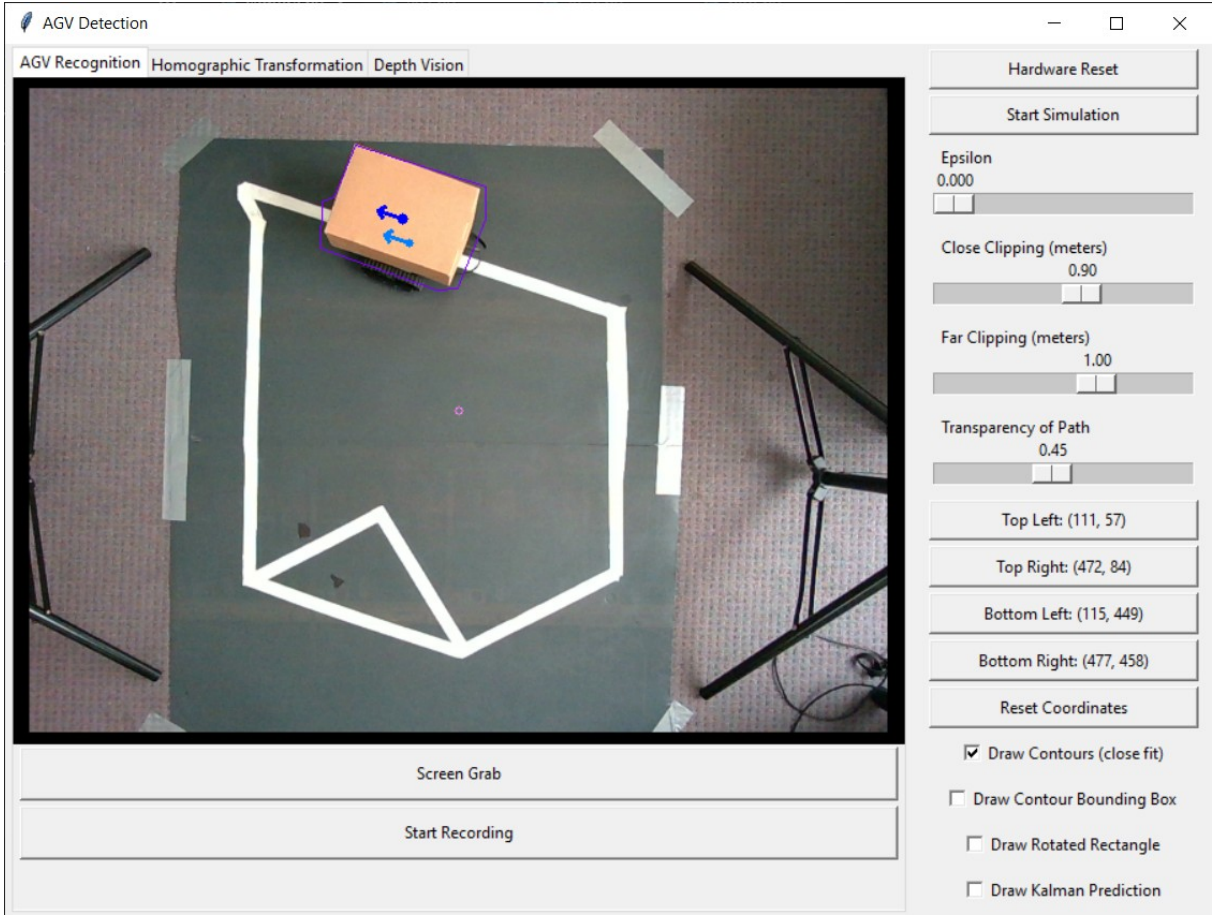


28 different modelling formalisms

50 transformations

FTG+PM: An Integrated Framework for Investigating Model Transformation Chains,
Levi Lúcio, Sadaf Mustafiz, Joachim Denil, Hans Vangheluwe, Maris Jukss.
Proceedings of the System Design Languages Forum (SDL) 2013, Montreal, Quebec.
Lecture Notes in Computer Science (LNCS), Volume 7916, pp 182-202, 2013.

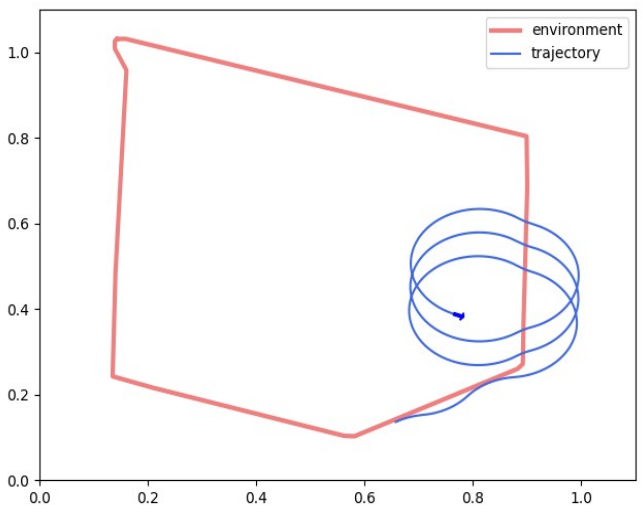
Line Following Robot (for Twinning research)



Design Iterations

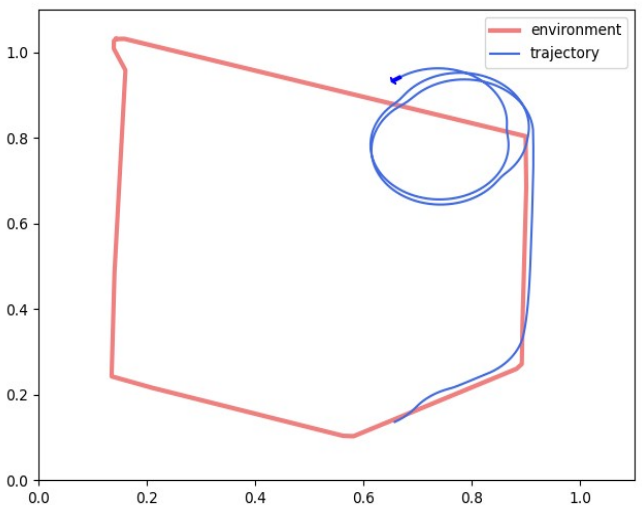
Initial Version

(Bang-Bang Controller with Centered Sensor)



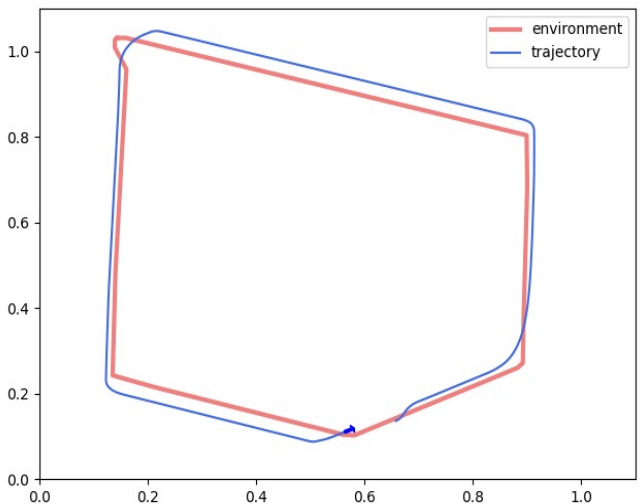
“fixed” Version

(Bang-Bang Controller with Offset Sensor)



“working” Version

(Tuned PID Controller with Offset Sensor)



Meta-Models (MM)

Formalism Transformation Graph (FTG)

Process Model (PM)

Process Trace (PT)

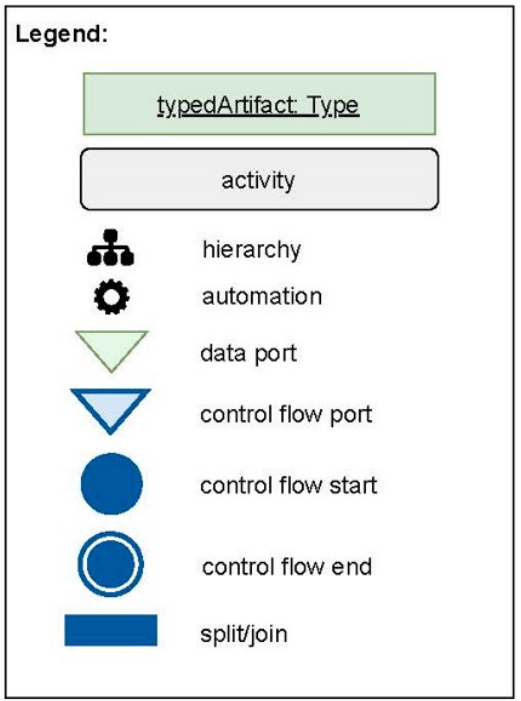
Storage, Services, Real-World Artifacts (S/S/RWA)

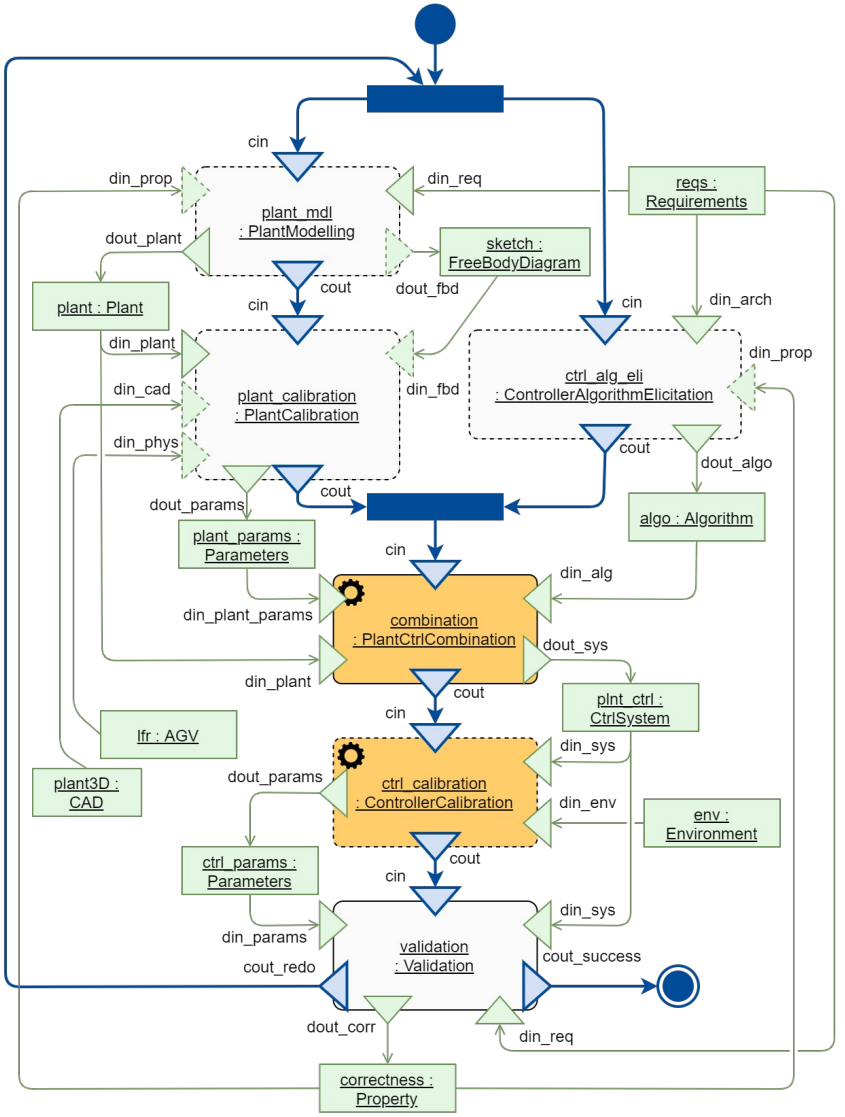
MM+FTG+PM+PT+S/S/RWA

aka FTG+PM++

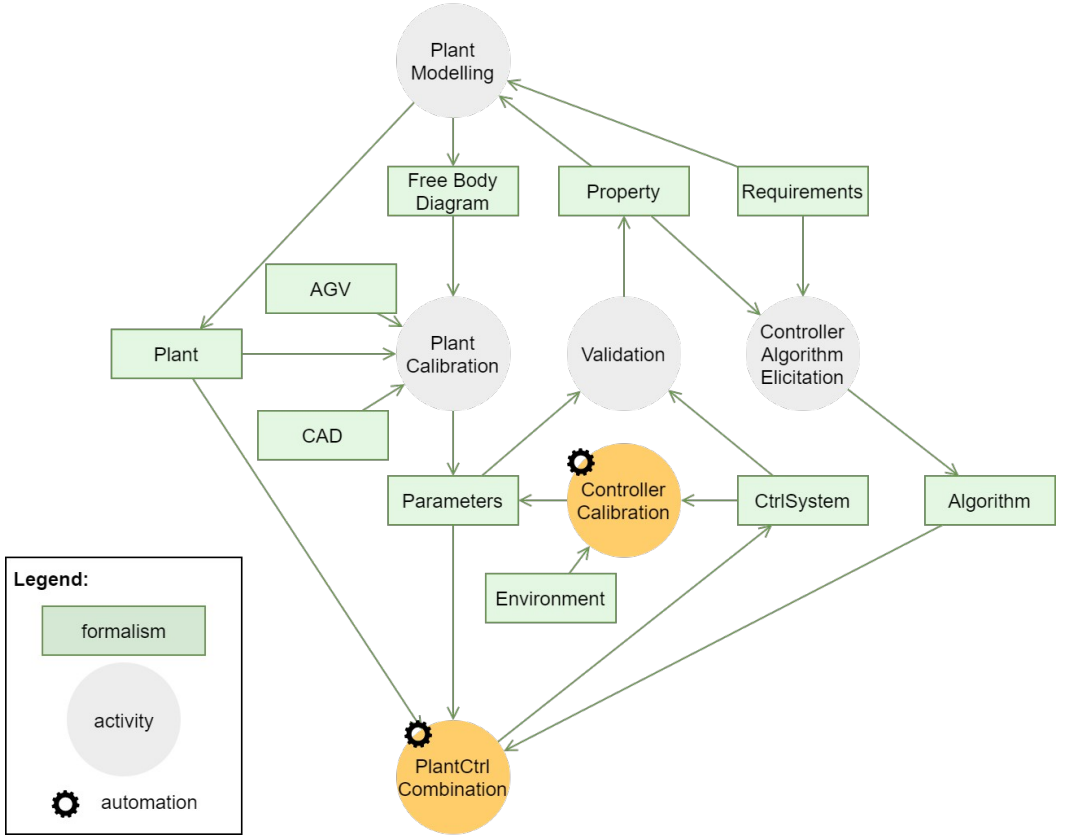
R. Paredis, J. Exelmans and H. Vangheluwe.
Multi-Paradigm Modelling For Model Based Systems Engineering: Extending The FTG + PM. 2022 Annual Modeling and Simulation Conference (ANNSIM), San Diego, CA, USA, 2022, pp. 461-474, doi: 10.23919/ANNSIM55834.2022.9859391.

Process Model

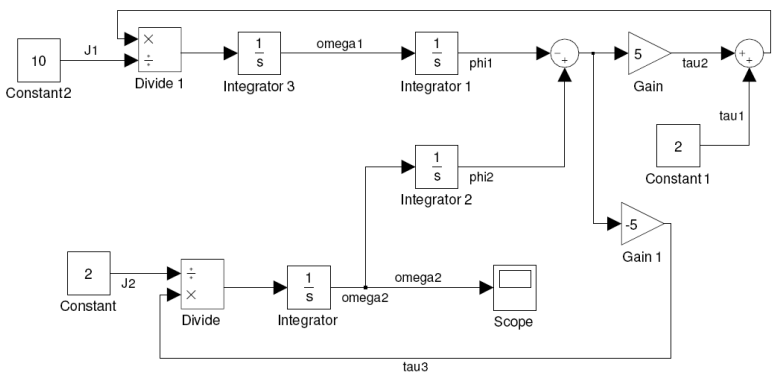


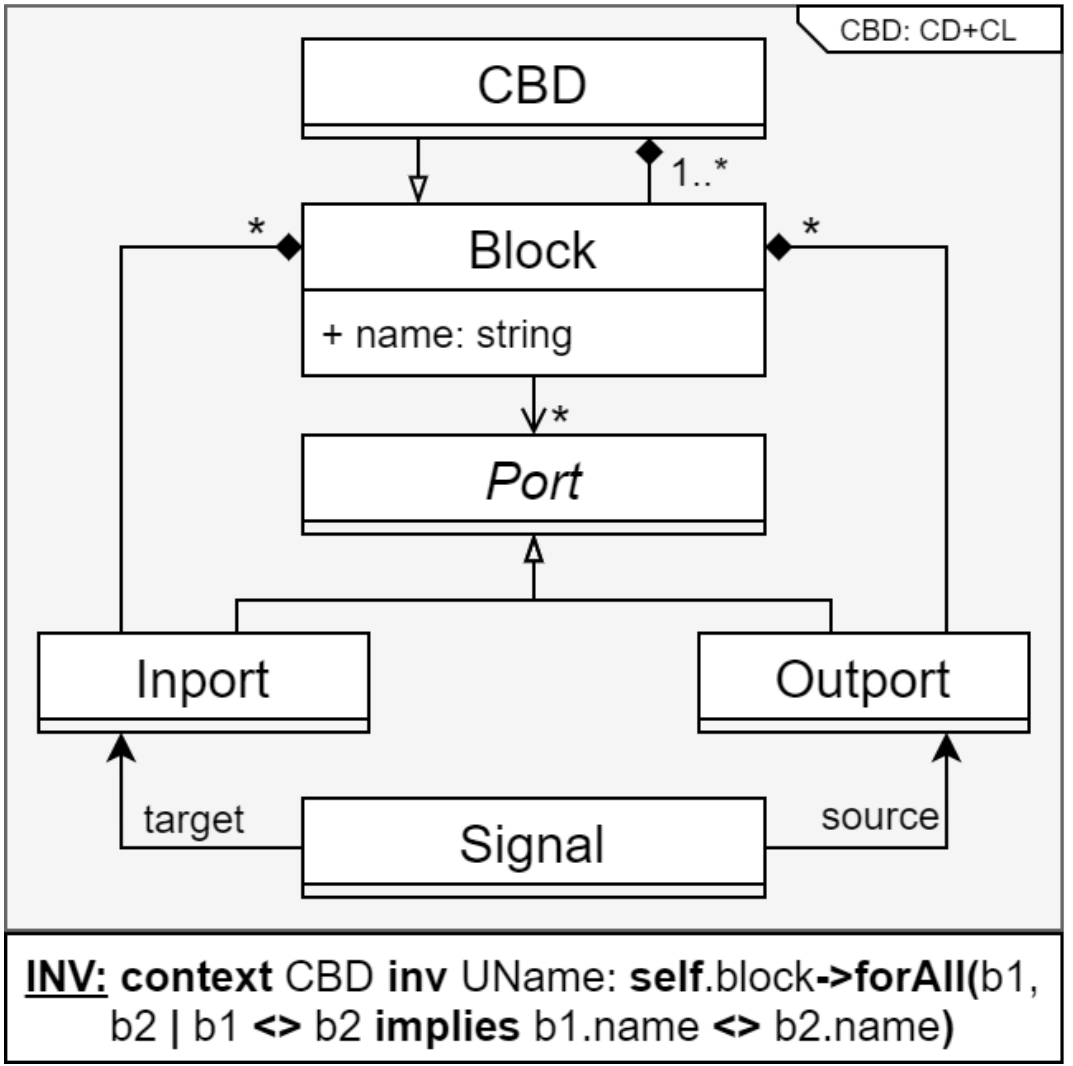


Formalism Transformation (R) Graph

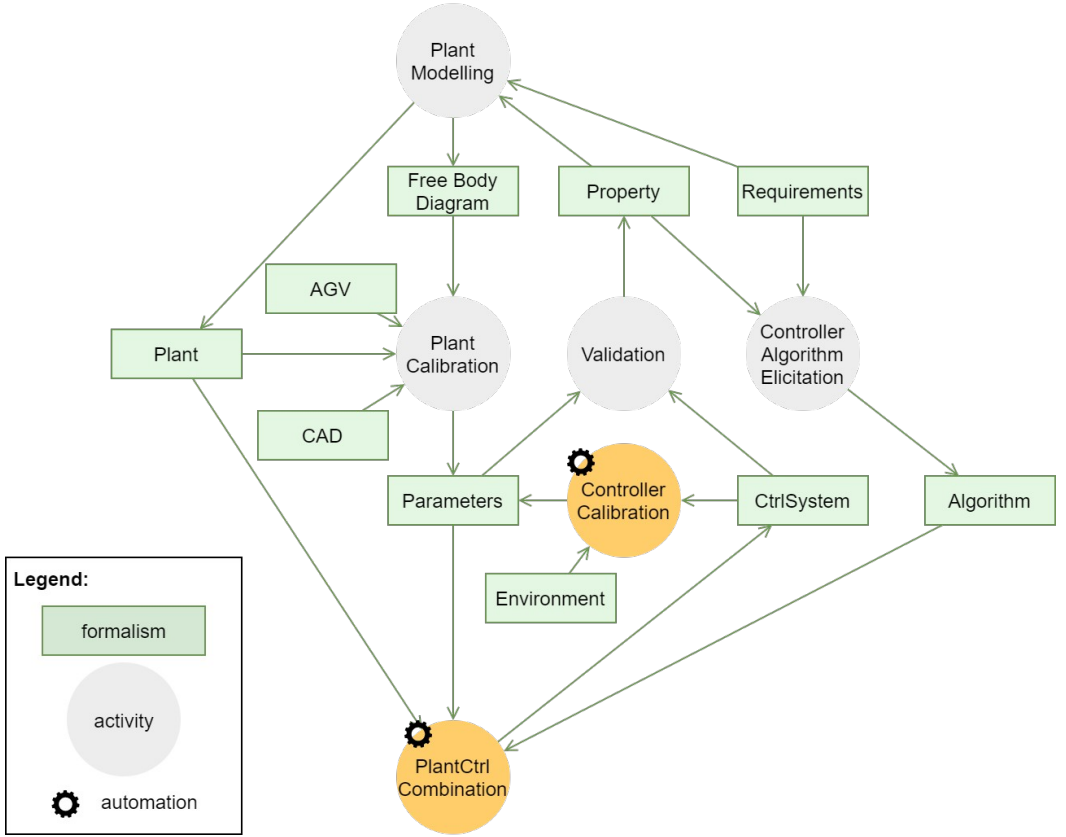


Meta-Models

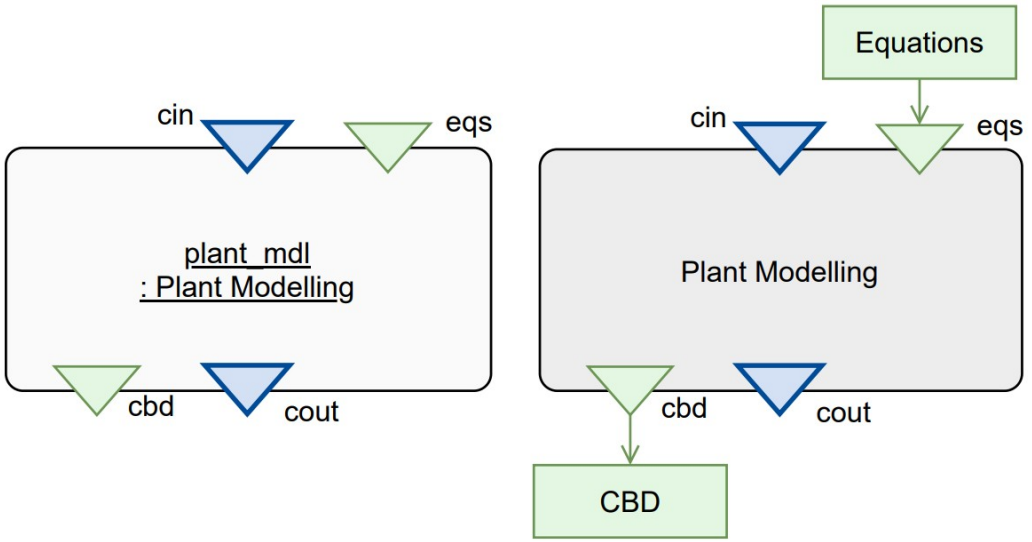

10
Constant2
J1
Divide 1
1/s
Integrator 3
omega1
1/s
Integrator 1
phi1
5
Gain
tau2
tau1
2
Constant1
1/s
Integrator 2
phi2
-5
Gain 1
2
Constant
J2
Divide
1/s
Integrator
omega2
omega2
Scope
tau3


CBD: CD+CL
CBD
1..*
*
Block
*
+ name: string
*
Port
Inport
Outport
target
Signal
source
INV: context CBD inv UName: self.block->forAll(b1, b2 | b1 <> b2 implies b1.name <> b2.name)

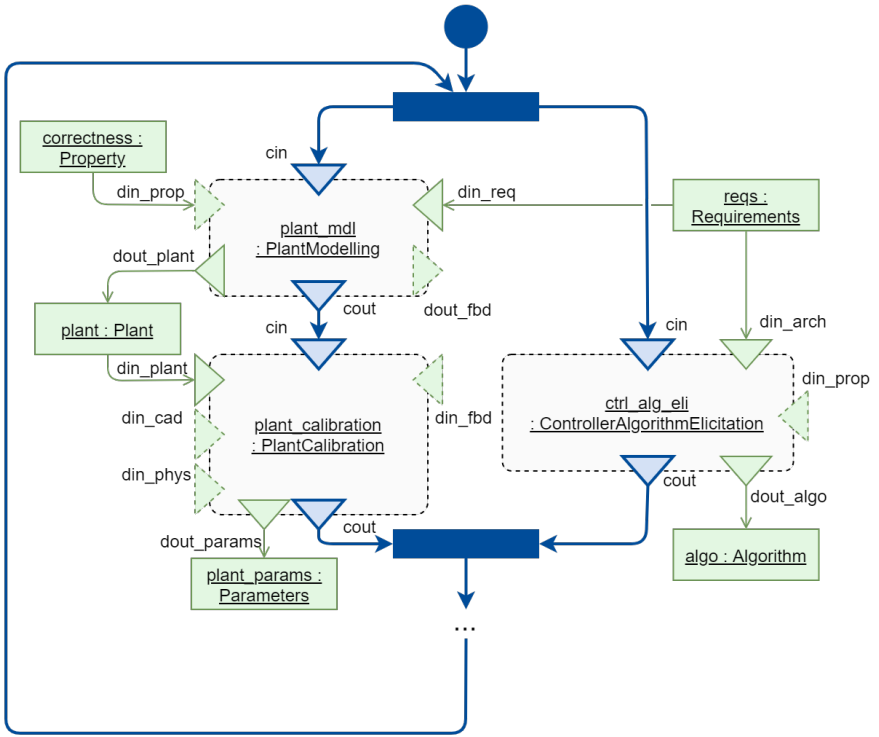
Formalism Transformation (R) Graph



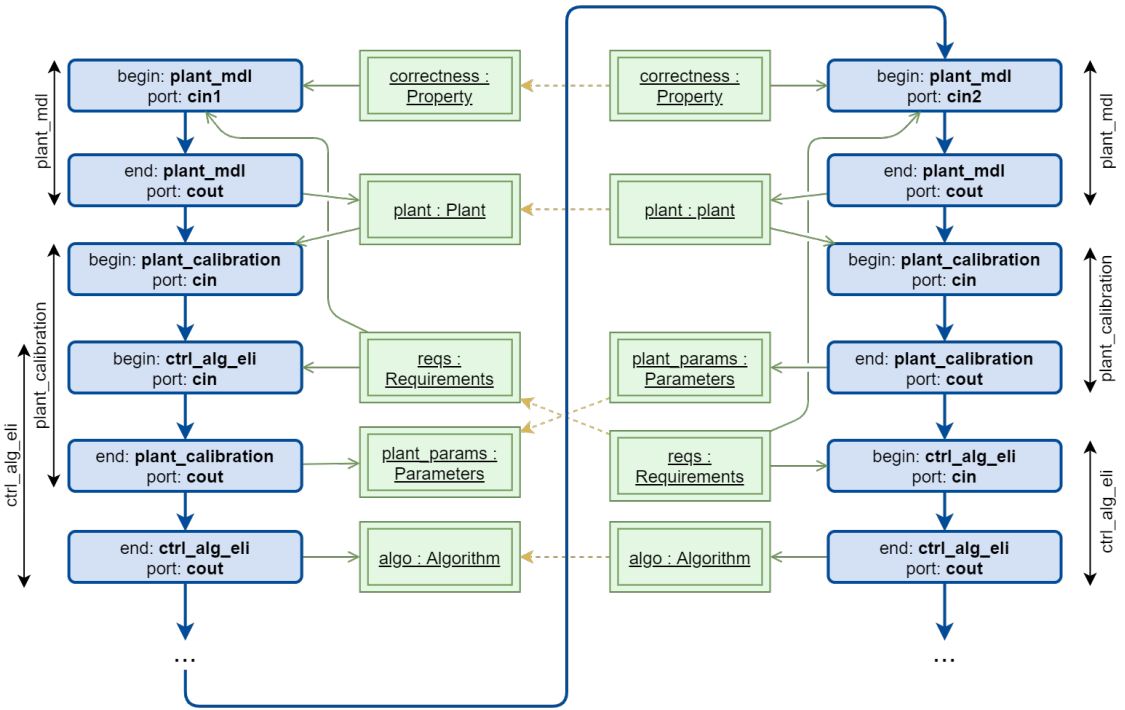
Activity Contracts



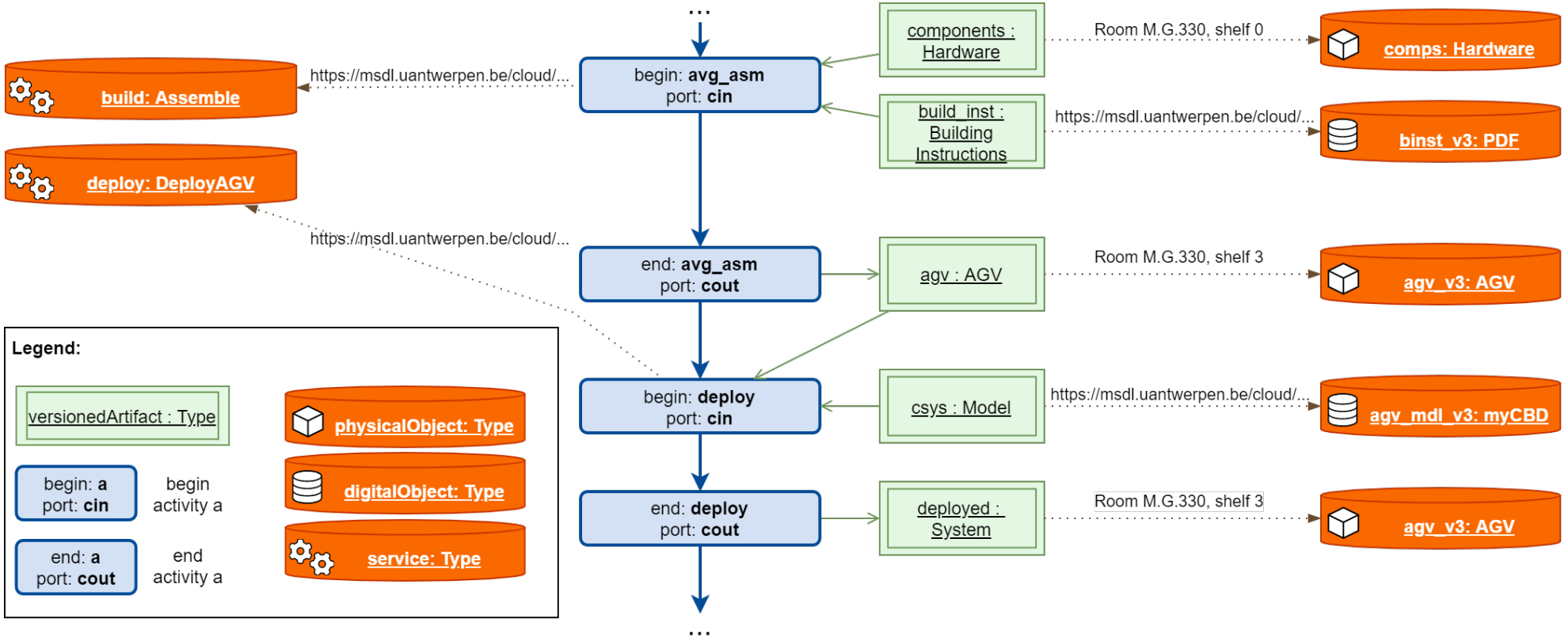
Process Model



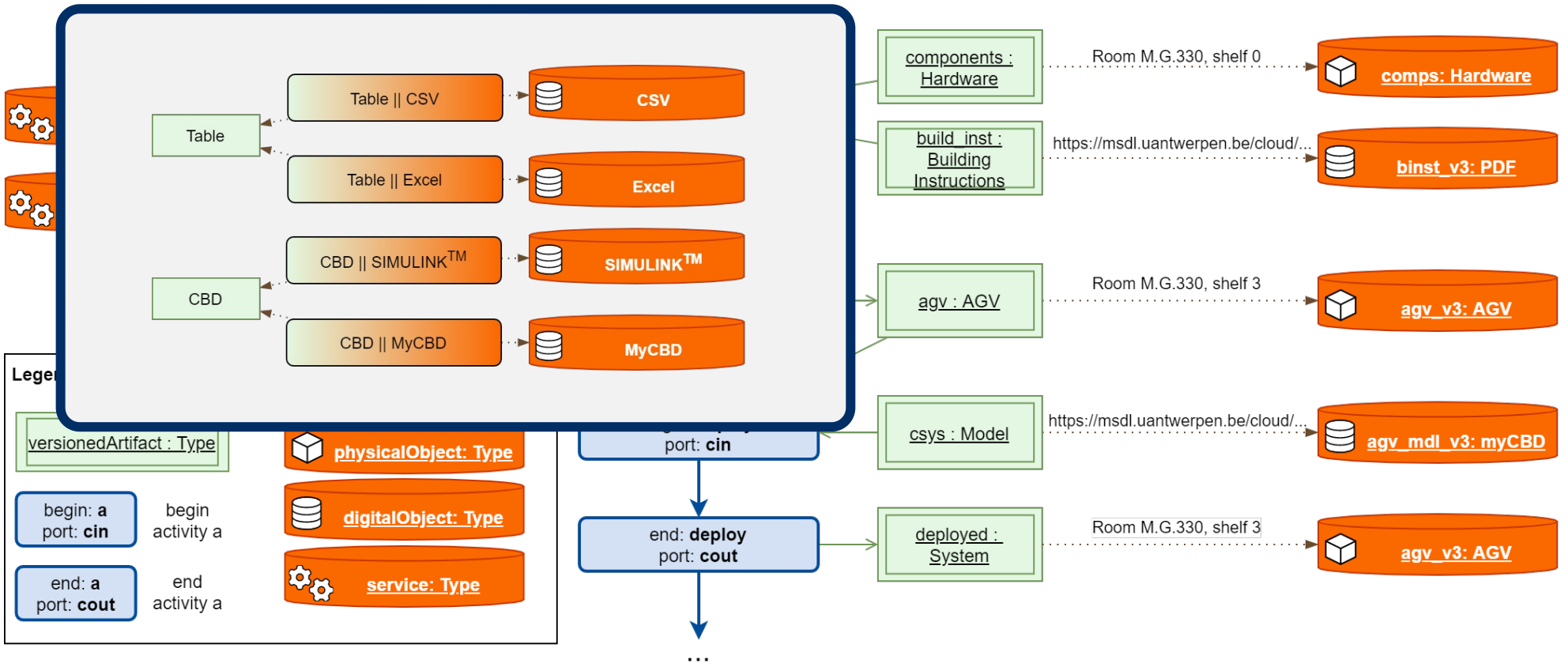
Process Trace

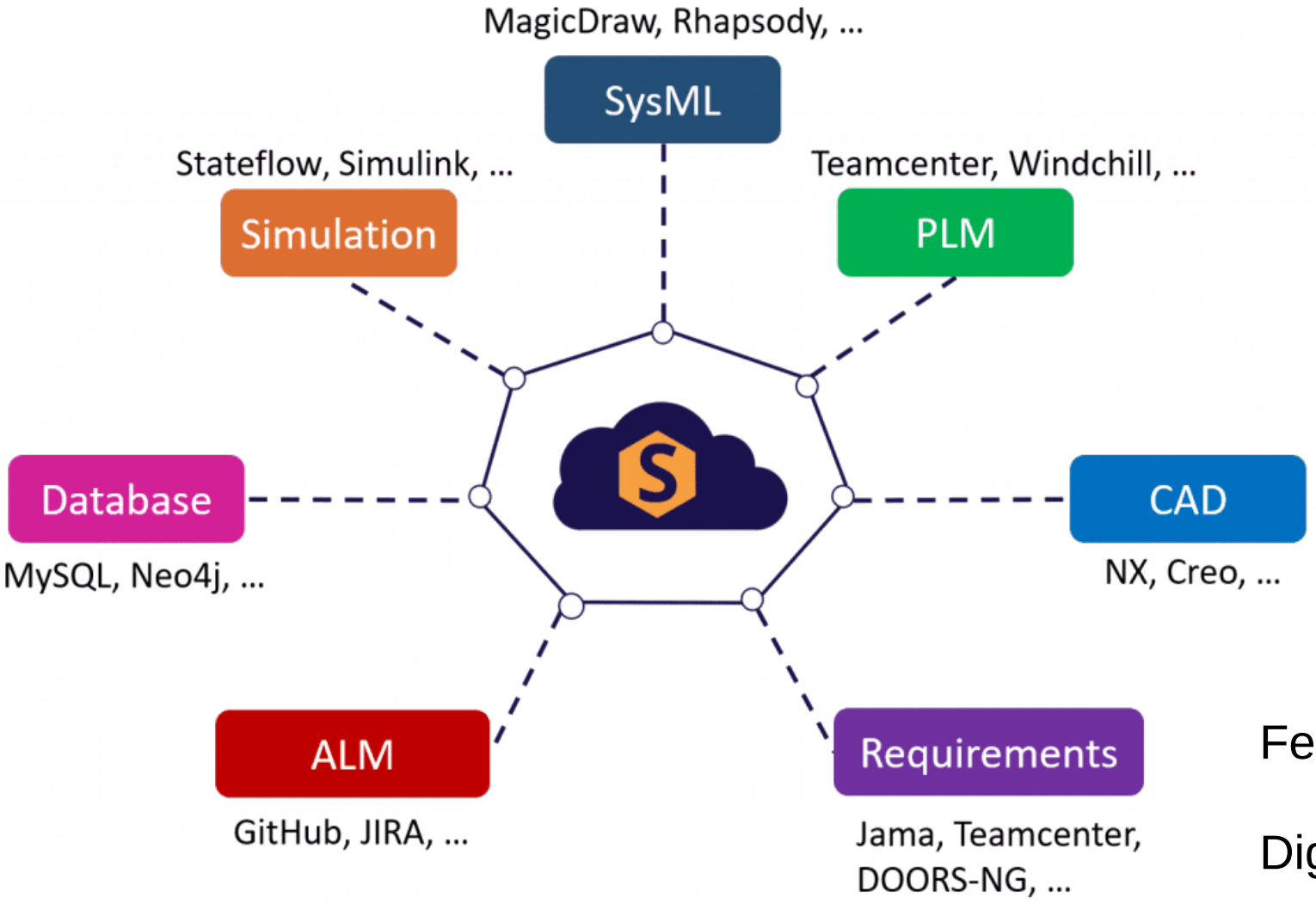


Adapters (Storage, Services, Real-World Artifacts)



Adapters (Storage, Services, Real-World Artifacts)

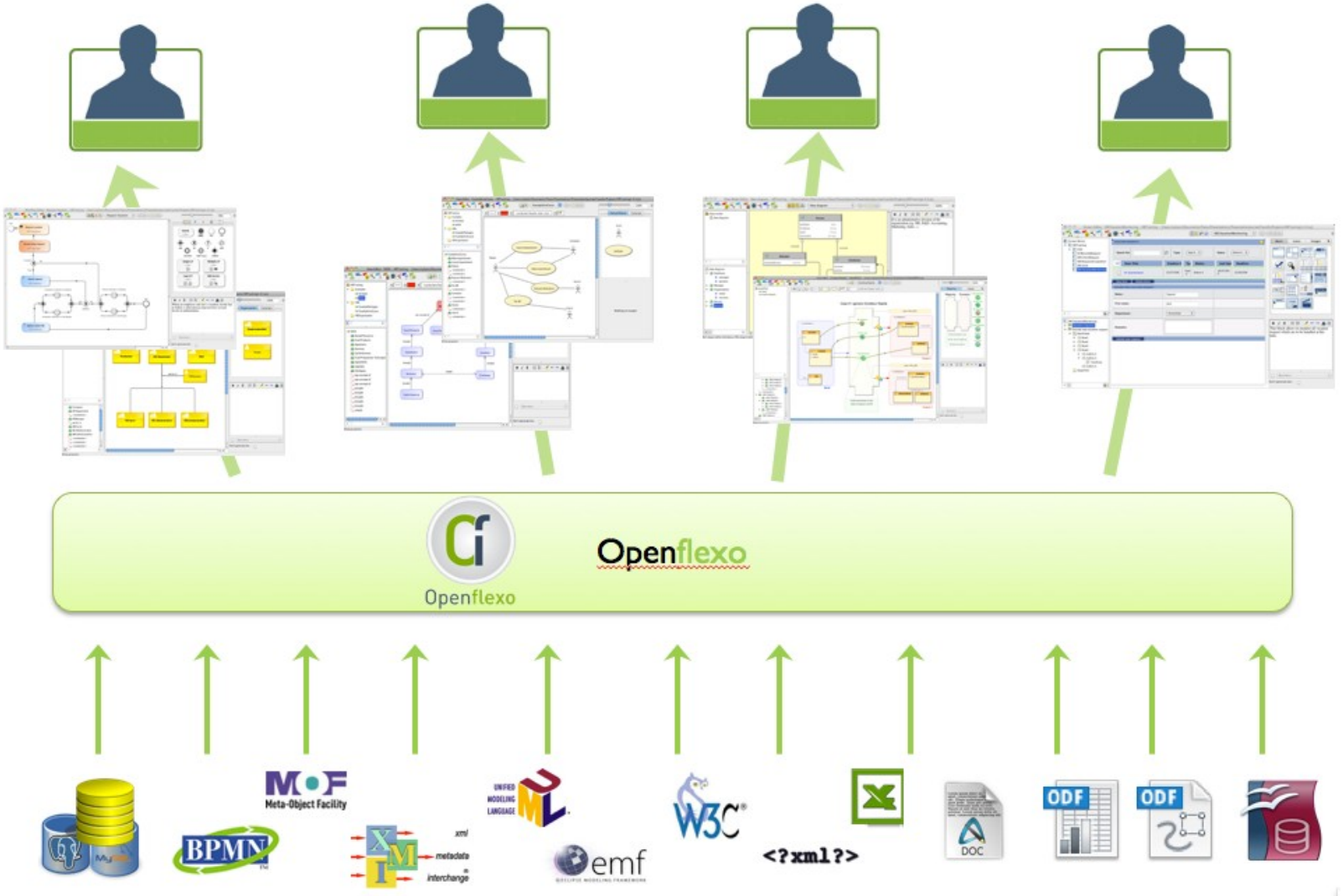




Federation (cfr. HLA)

Digital Thread

https://intercax.com/products/syndeia/



Sylvain Guerin

https://openflexo.org/

Types of Traceability (enabled by “model management”)

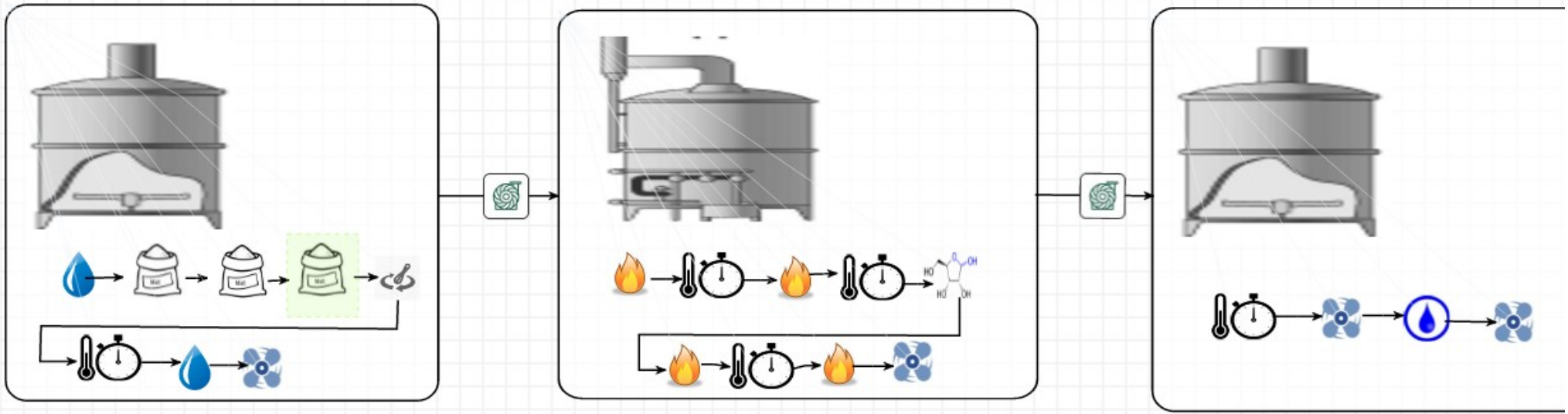
- Traceability linking experiment and system
- Traceability across artifact versions (and process model)
- Traceability based on properties of interest
- Traceability between artifacts on different levels of detail
- Traceability between instances and types
- Fine-grained traceability between artifact elements
- …

DSM TP 2014
Theory and Practice

5th International Summer School
on Domain Specific Modeling

Antwerp, Belgium
25 - 29 August

Thomas Kühne



Joachim Denil